Measurement of the $t\bar{t}$ Cross Section at the Run II Tevatron using Support Vector Machines

A dissertation

submitted by

Benjamin Eric Whitehouse

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in

Physics

TUFTS UNIVERSITY

August 2010

Advisor: Professor Krzysztof Sliwa

"Opportunity may knock, but it seldom nags."–David Mamet

Measurement of the $t\bar{t}$ cross section at the Run II Tevatron using Support Vector Machines

by Benjamin Eric Whitehouse

Advisor: Professor Krzysztof Sliwa

Abstract

This dissertation measures the $t\bar{t}$ production cross section at the Run II CDF detector using data from early 2001 through March 2007. The Tevatron at Fermilab is a $p\bar{p}$ collider with center of mass energy $\sqrt{s} = 1.96$ TeV. This data composes a sample with a time-integrated luminosity measured at 2.2 ± 0.1 fb^{-1}. A system of learning machines is developed to recognize $t\bar{t}$ events in the "lepton plus jets" decay channel. Support Vector Machines are described, and their ability to cope with a multi-class discrimination problem is provided. The $t\bar{t}$ production cross section is then measured in this framework, and found to be $\sigma_{t\bar{t}} = 7.14 \pm 0.25$ (stat) $^{+0.61}_{-0.86}$ (sys) pb.

Version 1.0

Acknowledgments

I would like to thank my advisor Professor Krzysztof Sliwa. He has allowed me the freedom to pursue my research interests, and was always an enthusiastic supporter of new ideas.

Also from Tufts, I would like to thank the high-energy physics professors. I have very fond memories of my courses, especially those of Larry Ford, Alan Everett, Alex Vilenkin, and Roger Tobin. Austin Napier was always happy to swap computer stories, and we had some good times while on shift at CDF. Simona Rolli was very helpful in getting me acclimated at FNAL, and she was always very willing to review my work. Finally, a big thank you to Jackie DiMichele and Jean Intoppa for arranging my travel logistics for these many years.

To my fellow graduate students and Sliwa advisees: Dan Ryan, Matt Hare, and Hao Sun. It has been great working with you over the past several years. We couldn't have all finished without the mutual support! Good work!

From CDF I would like to thank Kevin Lannon, who was ever ready to answer my questions. Steve Hahn was sympathetic to any PSM woes, and took care of any problems when I wasn't around.

Starbucks has undoubtedly been witness to the creation of the vast majority of these pages. Thanks to the North Beverly crew for making their store a welcome and friendly place conducive to writing about physics.

Barnes & Noble in downtown Naperville became a nice escape after a day in the lab. I must have read their entire drama and science fiction sections over the years. Thanks to Kerrilee for being a friendly face, ever ready with recommendations and a kind word.

To Dan Llacera, it has been great anticipating my eventual Florida visit. See you soon!

Mom and Dad, Jay, Ryan & Vanessa: you have been a great support to me for all this time. Thanks for your love and encouragement.

Andrew Beckwith, Bill Barker, Jacob Borgman:

Bill and Sara were my much needed home away from home (whether I happened to be in Illinois or England!). Thanks for all the visitations and explorations together.

Andrew and I have been able to share Patrick O'Brian's A&M over the past few years, and it is fitting that finally I'll be a real life doctor just as he is a real life captain. Though these words do not apply to either of us in the same way as in the novels, at least it's clear the link that exists between the two characters survives the superimposed translation from the book onto reality.

Jake and I have tackled many problems together throughout the course of my graduate and physics career. We explored SVMs, gravity, black holes, C++, generalized machine learning, and even the stock market. Jake, you've always stretched my horizons as others could not (heck, you even got me to buy a Mac!), and I'm very glad for our friendship.

Contents

Measurement of the $t\bar{t}$ cross section at the Run II Tevatron using Support Vector Machines

Part I

Theoretical Considerations

Chapter 1

The Standard Model

The major focus of this chapter is the summary introduction of the framework of high energy particle physics. It will serve as the jumping off point for a discussion of the fundamental components of the theory of Nature. Once complete, the top quark and its properties will be examined in slightly more detail, and the chapter will conclude with a discussion of the top quark's importance in the Standard Model.

1.1 Overview

The Standard Model of particle physics represents our most successful phenomenological description of Nature. It describes 3 of her 4 fundamental forces: the strong and weak nuclear forces, as well as electromagnetism. At this time gravity is described separately by the General Theory of Relativity. It remains one of the largest open problems in physics to combine the Standard Model and Relativity into a single Grand Unified Theory.

The Standard Model details the allowed interactions between particles through a specific application of Quantum Field Theory.

Excitations of these fields give rise to the particles of matter and their quantized force intermediators. What we call fermions are the particles which make up matter, while the bosons convey the interaction forces between fermions. Bosons and fermions are differentiated by the amount of intrinsic spin they carry: fermions possess 1/2 integer multiples of $\hbar$ ($\frac{n}{2}\hbar$, for n = 1,3,5...), while bosons have integer multiples ($n\hbar$, for n = 0,1,2,3...). Fermions are distinguished as either quarks or leptons, and each of *these* are further delineated in what are called generations. The main difference between

force	couples to	affects	boson	mass [GeV/c^2]	charge [e]	spin [$\hbar$]
EM	electric charge	charge carriers	photon (γ)	0	0	1
strong	color	quarks, gluons	gluon (g)	0	0	1
weak	weak charge	quarks, leptons, $W^{\pm}$, Z^0	$W^{\pm}$, Z^0	80.4 91.2	+1,−1 0	1
gravity	mass, energy	all	graviton (unobserved)	0	0	2

Figure 1.1: Fundamental Forces

particles across generations is in their mass, as every fermion of the 1^{st} generation has an analogous, heavier version in the 2^{nd} and 3^{rd}. Figure 1.1 enumerates the forces of the Standard Model, while Figure 1.2 provides a more detailed list of the properties of the fermions.

The allowable symmetries of the Standard Model are found according to the group $SU(3)_{QCD} \times SU(2)_L \times U(1)_Y$.

The strong interaction is governed by the $SU(3)_{QCD}$ gauge group. This force couples to the "color charge" of quarks, and comes in three varieties that are normally enumerated as red, green, and blue. The mediator bosons of the strong force are called *gluons*, which are massless and come in eight varieties derived from the allowable (color/anti-color) combinations. The theory of the strong interaction is thus aptly named Quantum ChromoDynamics (QCD).

The theory of the weak nuclear force has been successfully combined with that of electromagnetism in the "electro-weak" theory of Glashow, Salam, and Weinberg. The gauge group $SU(2)_L \times U(1)_Y$ of weak hyper-charge Y and weak isospin T governs its interactions.

Because the weak interaction only couples to left handed particles L, the fermion fields Ψ are split into left- and right-handed pieces $\Psi_{L,R} = \frac{1}{2}(1 \mp \gamma_5)\Psi$. These then give rise to left-handed doublet states with $T = \frac{1}{2}$, and right-handed singlet states where $T = 0$.

$$\begin{pmatrix} u \\ d \end{pmatrix}_L \quad \begin{pmatrix} c \\ s \end{pmatrix}_L \quad \begin{pmatrix} t \\ b \end{pmatrix}_L \quad \begin{matrix} u_R & c_R & t_R \\ d_R & s_R & b_R \end{matrix}$$

$$\begin{pmatrix} \nu_e \\ e \end{pmatrix}_L \quad \begin{pmatrix} \nu_\mu \\ \mu \end{pmatrix}_L \quad \begin{pmatrix} \nu_\tau \\ \tau \end{pmatrix}_L \quad \begin{matrix} \nu_{eR} & \nu_{\mu R} & \nu_{\tau R} \\ e_R & \mu_R & \tau_R \end{matrix}$$

The neutrinos and up-type quarks (u, c, t) of the doublets have isospin component $T_3 = +\frac{1}{2}$, while the down-type quarks (d, s, b) and charged leptons have $T_3 = -\frac{1}{2}$. Weak hyper-charge is defined to be $Y = 2Q - 2T_3$, where Q is electric charge. Hence, $Y = -1$ for leptons and $Y = \frac{1}{3}$ for quarks.

Non-zero masses for the particles of $SU(2)_L \times U(1)_Y$ violate the group's gauge symmetry. Thus, masses are introduced to the particles via the so-called "Higgs mechanism." This formal procedure introduces a complex scalar field doublet $\Phi = (\Phi^+, \Phi^0)^T$ to the Lagrangian. This spontaneously breaks the $SU(2)_L \times U(1)_Y$ symmetry to $U(1)_{QED}$ when the neutral component takes on a non-zero vacuum expectation value of $v/\sqrt{2}$. The three weak gauge bosons W^+, W^-, Z^0 acquire masses, while the QED boson (*i.e.* the photon) remains massless. The remaining degree of freedom from the introduced scalar doublet gives rise to a new particle in our framework, which is known as the Higgs boson. The Higgs is the last predicted Standard Model particle which remains to be observed experimentally.

The Particles of Matter

Fermion	1^{st} Generation	2^{nd} Generation	3^{rd} Generation	Q
Quarks	u (up)	c (charm)	t (top)	+2/3
	d (down)	s (strange)	b (bottom)	−1/3
Leptons	e^- (electron)	μ^- (muon)	τ^- (tau)	−1
	ν_e (electron neutrino)	ν_μ (muon neutrino)	ν_τ (tau neutrino)	0

Figure 1.2: Summary of Standard Model Fermions

The Higgs mechanism can also be used to give the fermions masses via Yukawa couplings to the Higgs scalar field. Through spontaneous symmetry breaking, the Higgs field gives rise to a mass for each fermion (m_f) which is proportional to the vacuum expectation: $m_f = (v\lambda_f)/\sqrt{2}$. Thus, each fermion has a free parameter (Yukawa coupling λ_f) which can be used to set its mass.

Because the mass eigenstates of electro-weak theory are not the same as the weak force eigenstates, it is possible for weak interactions (particle decays) to cross genera-

tions. The transformation of basis from the mass eigenstates to weak force eigenstates is encoded in the unitary matrix given in Equation (1.1), known as the CKM matrix (named for Cabibbo, Kobayashi, and Maskawa). As an example of its implications, weak processes like ($c \to W^+ s$) are the norm, while the small CKM off-diagonal elements also allow interactions such as ($c \to W^+ d$) to take place. If the CKM matrix were exactly equal to identity, such decays across quark generations would not be allowed.

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix}_L = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} d \\ s \\ b \end{pmatrix}_L \equiv V_{CKM} \begin{pmatrix} d \\ s \\ b \end{pmatrix}_L \tag{1.1}$$

$$\begin{pmatrix} |V_{ud}| & |V_{us}| & |V_{ub}| \\ |V_{cd}| & |V_{cs}| & |V_{cb}| \\ |V_{td}| & |V_{ts}| & |V_{tb}| \end{pmatrix} = \begin{pmatrix} (0.9741, 0.9756) & (0.219, 0.226) & (0.0025, 0.0048) \\ (0.219, 0.226) & (0.9732, 0.9748) & (0.038, 0.044) \\ (0.004, 0.014) & (0.037, 0.044) & (0.9990, 0.9993) \end{pmatrix} \tag{1.2}$$

fermion type	electric charge	Gen 1	Gen 2	Gen 3
Quarks	$+\frac{2}{3}$	up(u) 1.5 - 3.3 MeV/c^2	charm(c) 1.27 $^{+0.07}_{-0.11}$ GeV/c^2	top(t) 172.4 ± 1.2 GeV/c^2
	$-\frac{1}{3}$	down(d) 3.5 - 6.0 MeV/c^2	strange(s) 104 $^{+26}_{-34}$ MeV/c^2	bottom(b) 4.20 $^{+0.17}_{-0.07}$ GeV/c^2
Leptons	0	ν_e < 2 eV/c^2	ν_μ < 0.19 MeV/c^2	ν_τ < 18.2 MeV/c^2
	-1	e 0.511 MeV/c^2	μ 105.658 MeV/c^2	τ 1,777 MeV/c^2

Figure 1.3: Properties of Standard Model Fermions

Though originally thought to be massless, it has been determined that neutrinos also have some very small non-zero mass. The formalism of allowing for neutrino masses gives rise to a similar basis mixing, which in turn allows for transitions across lepton generations in analogy to the CKM matrix. This matrix is called the PMNS matrix (for Pontecorvo, Maki, Nakagawa, and Sakata).

In summary, the Standard Model[2] uses Quantum Field Theory to describe Nature. It requires a total of 25 free parameters which must come from empirical measurements.

These are:

- 2 parameters for the Higgs Mechanism (m_H, v)
- 3 force coupling constants (α_s, g, g')
- 4 parameters of CKM matrix
- 4 parameters of PMNS matrix
- 12 Yukawa couplings that give mass to the fermions

1.2 The Top Quark

The top quark was discovered in 1994 here at the Tevatron during Run I. A nice summary of that time period is given by Liss and Tipton in *Scientific American*[5]. Top quark pair production can occur through quark/anti-quark annihilation or gluon-gluon fusion during the $p\bar{p}$ collision. Figure 1.4 shows leading order Feynman diagrams for the possible configurations.

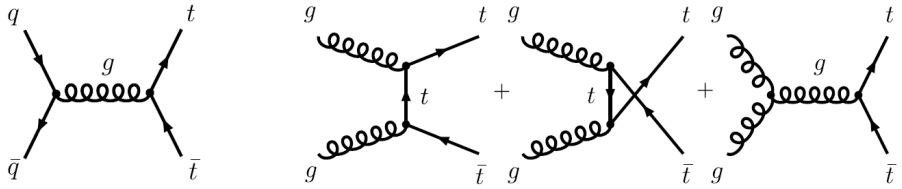


Figure 1.4: Feynman Diagrams for $t\bar{t}$ pair production.

As can be seen from the magnitudes of the last row of the CKM matrix listed in Equation (1.2), the dominant decay mode for the top quark is into the two body system ($t \to W^+b$). Decays to both W^+s and W^+d are highly suppressed, and are not considered further in this thesis. The b quark in this decay leads to a hadronic jet, while the W boson can decay hadronically into quarks, or leptonically into a lepton/neutrino pair. The $t\bar{t}$ decays therefore fall into three different channels. The dilepton channel (DIL), represents the case where both W's decay leptonically. The lepton plus jets channel (L+J) has one of the W's decay leptonically and the other hadronically. The all hadronic (HAD) encompasses all cases where both W's turn into hadronic jets. See Figure 1.5.

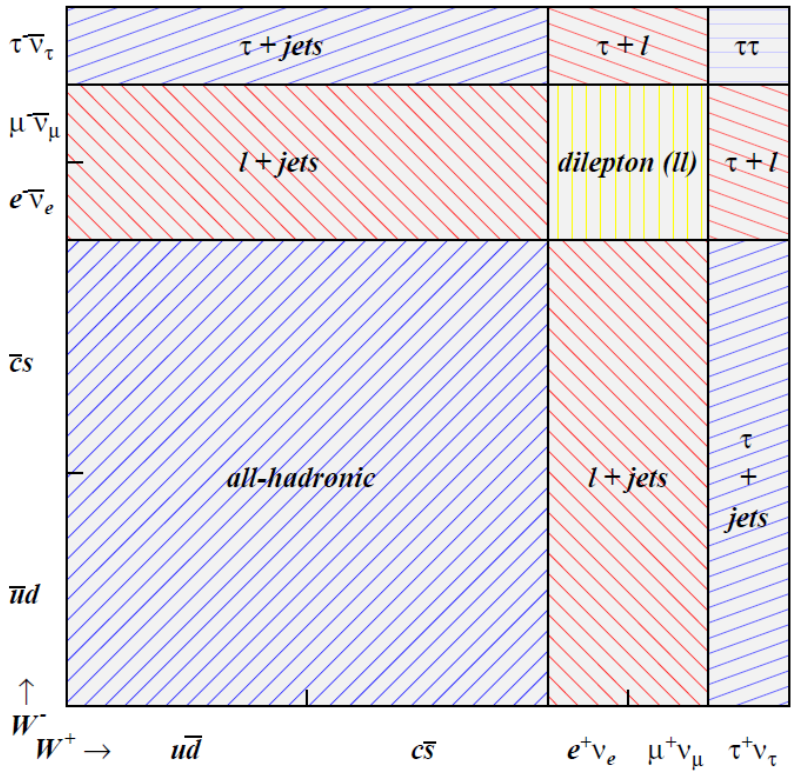


Figure 1.5: $t\bar{t}$ Decay Channels

The $t\bar{t}$ pair production cross section ($\sigma_{t\bar{t}}$) comes primarily from valance quark annihilation between the proton and anti-proton. Only about 15% of $\sigma_{t\bar{t}}$ comes from gluon-gluon fusion. This is due to the fact that the center of mass energy of the Tevatron ($\sqrt{s} = 1.98$ TeV) is relativly close to the production threshold for $t\bar{t}$ because of the quark's large mass. Two theoretical calculations for the $t\bar{t}$ cross section are quoted in Figure 1.6. Both of the values listed in this figure use the CTEQ6.6 PDF set at NNLO, and assume a top mass $m_t = 175$ GeV/c^2. See [25], [26], [27], and [28] for other assumed top masses and/or PDFs.

$\sigma_{t\bar{t}}$ [pb]	Reference
$6.61^{+0.51}_{-0.58}$	Cacciari *et al.*[26]
$6.73^{+0.51}_{-0.46}$	Kidonakis *et al.*[28]

Figure 1.6: Theoretical $\sigma_{t\bar{t}}$ Calculations

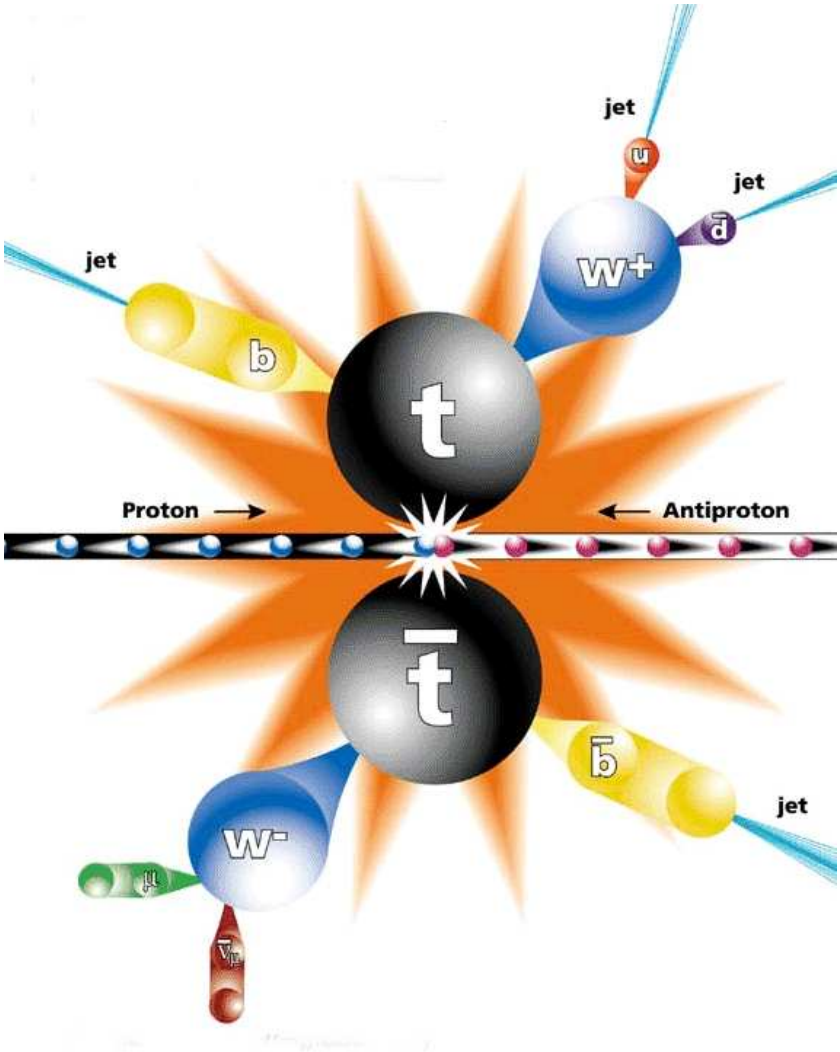


Figure 1.7: Cartoon of L+J $t\bar{t}$ Production

Part II

SVM Theory

Chapter 2

The Support Vector Machine

The overall idea for the Support Vector Machine was put forth by Vladimir Vapnik[20] in the 1990s. Using labeled training data as a basis, the SVM methodology attempts to calculate the optimal separating hyperplane between the two classes of data under consideration. In order to tell apart apples from oranges, one might consider any number of descriptive, quantifiable features of these two objects. Color, texture, size, lifetime, ripening time, seed content, skin width, shape, density, mass, *etc.* can all be assembled in a vector of values used to describe an apple or an orange. Given a labeled handful of these vectors, the SVM formalism computes a separating hyperplane that attempts to place all the apples on one side and all oranges on the other. The purpose of this chapter is to set forth the theory behind Support Vector Machines, point out some of their interesting properties, and show how they can be applied to larger classification problems.

2.1 The Basics

Suppose that we define a vector of real numbers, where each dimension represents some characteristic of the subject we wish to study. The N dimensions of this vector are known as *features*. Let us define

$$\vec{x} \equiv \{x_1, x_2, \ldots, x_N\} \tag{2.1}$$

$$y \equiv \pm 1$$

We also need a way to distinguish between the two classes of data. We will use the convention that every data point is specified by its vector $\vec{x}$, and its class y when this information is available.

Imagine we are given some number of events whose class is known. Some of the points belong to the $y = +1$ class, and others to the $y = -1$ class. Suppose we plot these vectors in real N dimensional space in the normal way. Our task is to then use these event vectors to find a separating hyper-plane between the points with $y = +1$ and $y = -1$. For vectors with only 2 features, consider Figure 2.1. As can be seen, it is necessary to establish a way to uniquely define the plane of separation. The method that is chosen is to find the plane that separates the two classes of points with the widest *margin*. Figure 2.2 defines the geometry of the situation in a two feature problem.

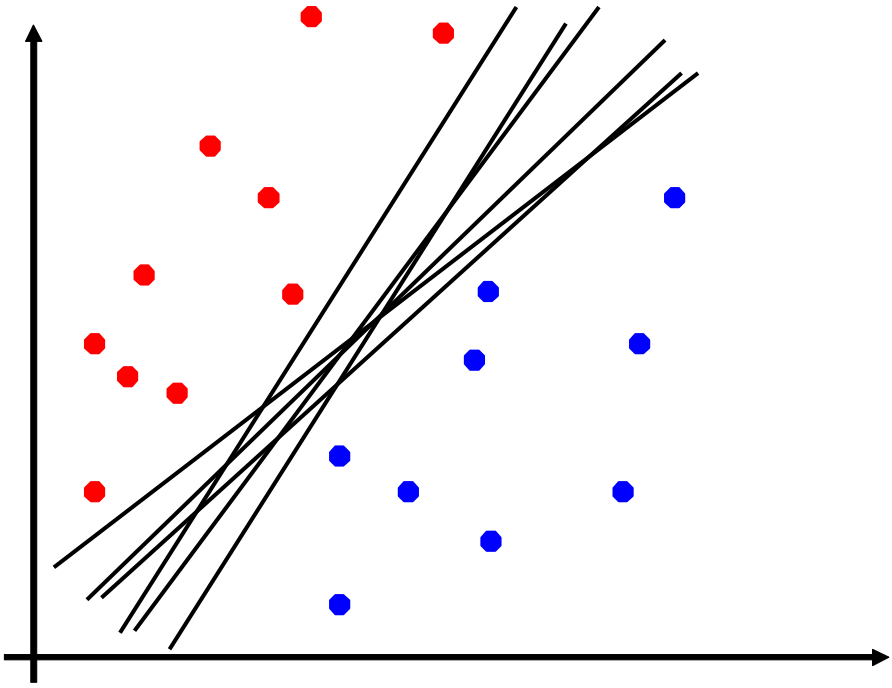


Figure 2.1: Many planes could potentially separate our data.

For an N dimensional space, the equation of a hyper-plane in that space is given by:

$$f(\vec{x}) = \vec{w} \cdot \vec{x} - b \tag{2.2}$$

where $\vec{w}$ is a vector normal to the plane, and b is a real number that offsets the plane's

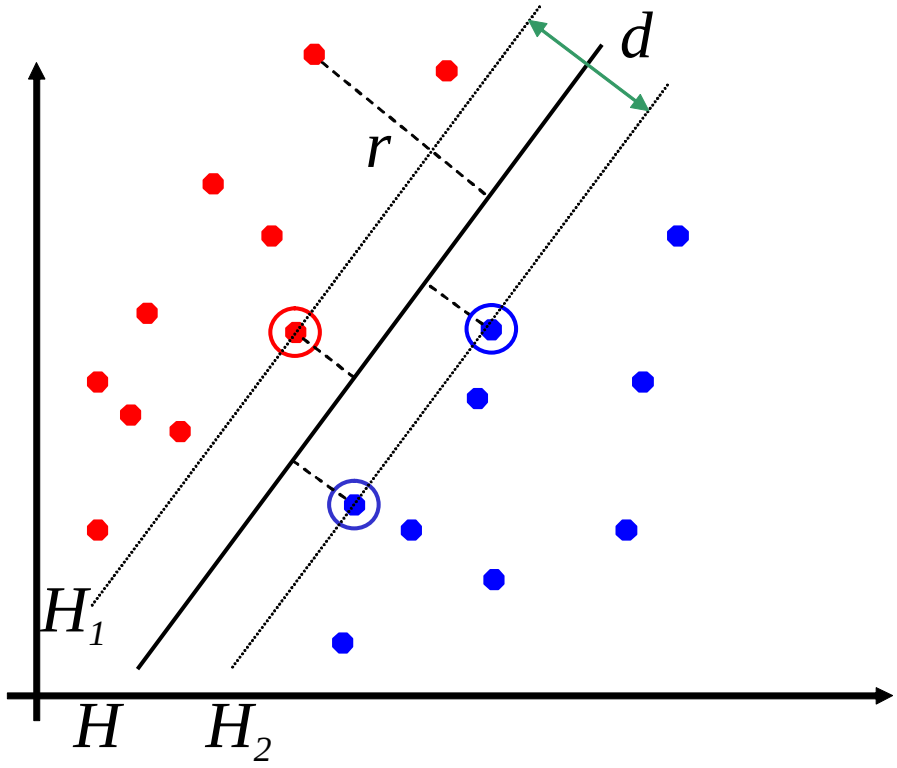


Figure 2.2: 2D SVM Geometry

position from the origin. Once $\vec{w}$ and b have been determined, this expression becomes the *learned function*. After computing $f(\vec{a})$ on some vector of unknown class, the sign of $f(\vec{a})$ can be our guess as to which class $\vec{a}$ belongs.

Looking more closely at the learned function, we see that the perpendicular distance r from the plane to a point $\vec{z}$ is given by

$$r = \frac{\vec{w} \cdot \vec{z} - b}{|\vec{w}|} \tag{2.3}$$

A further step needs to be taken to set the overall scale for the problem. This is because there are an infinite number of vectors $\vec{w}$ that give the same orientation for the plane. We could simply constrain $\vec{w}$ to be a unit vector. However, a more convenient method for our future calculations is simply to compute $\vec{w}$ such that the value of the learned function is ± 1 at the margin. This fixes the scale, and the overall width of the margin is then

$$d = \frac{2}{|\vec{w}|} \tag{2.4}$$

This quantity is clearly maximized if we minimize the length of $\vec{w}$. To be clear, we have now totally specified our geometry. $\vec{w}$'s direction gives the orientation of the separating hyperplane, b displaces it from the origin to its proper absolute position, and $\vec{w}$'s magnitude sets the width of the margin (*i.e.* the density of the contours parallel to the hyper-plane).

In order to ensure that no data points invade the margin, we also require the following linear constraints on each of the N training points under consideration.

$$y_i(\vec{w} \cdot \vec{x_i} - b) \geq 1 \tag{2.5}$$

Putting everything together, we can assemble a Lagrangian to minimize for this problem.

$$\mathscr{L} = \frac{1}{2}\vec{w} \cdot \vec{w} - \sum_{i=1}^{N} \alpha_i y_i(\vec{w} \cdot \vec{x_i} - b) + \sum_{i=1}^{N} \alpha_i \tag{2.6}$$

The first term maximizes the margin, and the subsequent N terms ensure the separation of the two classes in our training set. Notice the use of Lagrange multipliers ($\alpha_i' s \geq 0$) to enforce these linear constraints. This approach is common in solving quadratic optimization problems such as we have here. A dual Lagrangian can be constructed by taking the partial derivatives with respect to the primal variables $\vec{w}$ and b and noting they are zero at optimality. This leads to the following relations

$$\frac{\partial \mathscr{L}}{\partial \vec{w}} = 0 \Rightarrow \vec{w} = \sum_{i=1}^{N} \alpha_i y_i \vec{x_i} \tag{2.7}$$

$$\frac{\partial \mathscr{L}}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} \alpha_i y_i = 0 \tag{2.8}$$

Substituting these back into our original Lagrangian, we get the dual Lagrangian

$$\mathscr{L}_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \vec{x_i} \cdot \vec{x_j} \tag{2.9}$$

This formulates our problem entirely in terms of the α's. Once the α's are known, we can recover $\vec{w}$ through the relation (2.7) above, and b from the fact that

$$b = \vec{w} \cdot \vec{x_k} - y_k \tag{2.10}$$

for any training vector k, as long as its corresponding $\alpha_k \neq 0$. Using our new formulation we can re-express the learned function in terms of the α's.

$$f(\vec{z}) = \sum_{i=1}^{N} \alpha_i y_i \vec{x_i} \cdot \vec{z} - b \tag{2.11}$$

The important thing to note here is that our learned function depends only on those training points $\vec{x_i}$ which have an $\alpha_i \neq 0$. Training points that meet this requirement are known as the *Support Vectors*, as they are the vectors that hold up the decision plane. Only support vectors, along with their associated Lagrange multipliers, are needed to reconstruct the decision plane. All other training points are ignorable. Also, notice that the learned function only depends on the inner product between training points and the test point $\vec{z}$. We will return to this later when considering non-linear decision surfaces.

2.2 Mechanical Analogy

An interesting interpretation of the formulation of the Support Vector Machine reveals it to have an exact physical analog. The support vectors can be viewed as exerting a positive or negative force on the decision boundary, and the solution in which we find the best separating hyper-plane is exactly that which provides mechanical equilibrium. In other words, all we are requiring is that the sum of all the forces and torques net zero at optimality. This view can be helpful in getting a handle on what actually is happening when a SVM is optimized. It can be useful to think about how certain points "push around" the decision boundary.

To illustrate this point, let's associate a force $\vec{F}_i$ with each support vector. We will take

$$\vec{F}_i = -y_i \alpha_i \hat{w} \tag{2.12}$$

Here, $\hat{w}$ is simply a unit vector in the direction of $\vec{w}$, defined as $\hat{w} = \frac{\vec{w}}{|\vec{w}|}$. Recall from Equation (2.2) that $\hat{w}$ gives the orientation of the hyper-plane. In fact, $\hat{w}$ is normal to the hyper-plane's surface. Therefore, Equation (2.12) associates a force normal to the hyper-plane which points inwards toward the zero contour of the decision surface. The

magnitude of the force is proportional to the size of that vector's Lagrange multiplier α_i. The requirements for mechanical equilibrium are as follows:

$$\sum_{i}^{SVs} \vec{F}_i = 0 \tag{2.13}$$

$$\sum_{i}^{SVs} \vec{x}_i \times \vec{F}_i = 0 \tag{2.14}$$

Using Equations (2.7) and (2.8), we see that these conditions hold.

$$\sum_{i}^{SVs} \vec{F}_i = \sum_{i}^{SVs} -y_i \alpha_i \hat{w} = -\hat{w} \sum_{i}^{SVs} y_i \alpha_i = 0 \tag{2.15}$$

$$\sum_{i}^{SVs} \vec{x}_i \times \vec{F}_i = \sum_{i}^{SVs} \vec{x}_i \times (-y_i \alpha_i)\hat{w} = \sum_{i}^{SVs} (-y_i \alpha_i)\vec{x}_i \times \hat{w} = -\vec{w} \times \hat{w} = 0 \tag{2.16}$$

2.3 Soft Margin

Sometimes, it happens that a problem is not perfectly separable. Imagine there is some noise in the problem, or that some points in the training sample are mislabeled. To accommodate this kind of situation, our problem can be reformulated to allow some points to invade the margin. However, we will penalize these points in order to discourage their occurrence. To relax the margin constraints, we modify Equation (2.5) to read

$$\begin{aligned} y_i(\vec{w} \cdot \vec{x_i} - b) &\geq 1 - \xi_i \\ \xi_i &\geq 0 \end{aligned} \tag{2.17}$$

These free parameters ξ_i allow individual points to enter the margin. However, if we don't suppress this behavior in some manner, then the trivial solution where all the points fall into the margin will result. So, we modify the Lagrangian as follows:

$$\mathscr{L} = \frac{1}{2}\vec{w} \cdot \vec{w} + C \sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} \alpha_i y_i (\vec{w} \cdot \vec{x_i} - b) + \sum_{i=1}^{N} \alpha_i (1 - \xi_i) \tag{2.18}$$

The second term in (2.18) involving C is the penalty term. Since we are minimizing $\mathscr{L}$, and both C and the ξ_i's are taken to be non-negative, this term will tend to cap the error. It is instructive to view the Lagrangian rearranged like this:

$$\mathscr{L} = \frac{1}{2}\vec{w} \cdot \vec{w} - \sum_{i=1}^{N} \alpha_i y_i (\vec{w} \cdot \vec{x_i} - b) + \sum_{i=1}^{N} \alpha_i + \sum_{i=1}^{N} \xi_i (C - \alpha_i) \tag{2.19}$$

Equation (2.19) has exactly the same form as (2.6), except for the last term that isolates ξ. In order to constrain $\mathscr{L}$, we note what happens to the objective function when α's are allowed to grow larger then C. In that case, the sum in the last term can be driven to $-\infty$ by letting the ξ's become arbitrarily large. To avoid this situation, we require that the α's are bound within the range $0 \leq \alpha_i \leq C$. This result follows naturally by taking

$$\frac{\partial \mathscr{L}}{\partial \xi_i} = 0 \Rightarrow \alpha_i = C \tag{2.20}$$

for any ξ_i that exists (that is $\xi_i > 0$). A very convenient by product of applying this constraint is that it eliminates the last term in (2.19)! We are left with exactly Equation (2.6), and the rest of the solution follows as before. Equations (2.7) and (2.8) are the same, as is the dual $\mathscr{L}_D$ in (2.9). We are solving exactly the same problem as previously, except we have restricted the range of the Lagrange multipliers. We have a new free parameter C that controls the rigidity of the margin. To summarize, the problem we are solving is

$$\begin{aligned} \mathscr{L}_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \vec{x_i} \cdot \vec{x_j} \\ 0 \leq \alpha_i \leq C \\ \sum_{i=1}^{N} \alpha_i y_i = 0 \end{aligned} \tag{2.21}$$

Note that there is also a formulation that fixes the overall invasion "length" by placing an upper bound on the sum of the ξ variables. This is in contrast to the method just outlined which limits the individual penetration of the given training points, but places no bound on the number of points which invade the margin.

2.4 Non-linear Solutions

Up to this point, we have shown how to find the best separating hyper-plane for a given set of training data. However, the hyper-plane we have constructed exists in the same space as the data points, or what is known as the *feature space*. One drawback to this situation is that it may not be possible to find a separating plane for the problem we would like to solve. It would be nice if we were able to carry our support vector formalism over to such problems that only have non-linear solutions. This can be accomplished if we map our original space into some other feature space of higher dimensionality. The hyper-plane can then be constructed in this higher dimensional space where the problem is linearly separable.

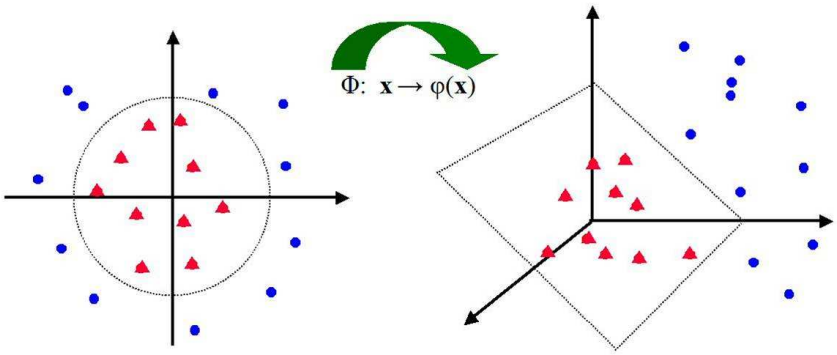


Figure 2.3: Mapping problem into higher dimensional space

Notice that in Equations (2.21) and (2.11) the training vectors $\vec{x_i}$ only enter the calculations when being dotted into another $\vec{x_j}$. If we were to actually do a mapping on the original vectors $\vec{x_i} \Rightarrow \varphi(\vec{x_i})$, the only expressions that show up in the training procedure are of the form $\varphi(\vec{x_i}) \cdot \varphi(\vec{x_j})$. Therefore, as long as we can carry out the dot products in the higher dimensional space, *we need not know the exact details of the mapping function* φ.

It turns out that there is a whole class of functions, known as *kernel functions*, which are dot products in some multi-dimensional space. In fact, any positive semi-definite function that can be shown to have the form given in (2.22) is a kernel function. Equations (2.23) through (2.26) are some example kernel functions:

$$K(\vec{x_i}, \vec{x_j}) \equiv \varphi(\vec{x_i}) \cdot \varphi(\vec{x_j}) \tag{2.22}$$

$$K(\vec{x_i}, \vec{x_j}) = \vec{x_i} \cdot \vec{x_j} \tag{2.23}$$

$$K(\vec{x_i}, \vec{x_j}) = (1 + \vec{x_i} \cdot \vec{x_j})^p \tag{2.24}$$

$$K(\vec{x_i}, \vec{x_j}) = \tanh(\beta_0 \vec{x_i} \cdot \vec{x_j} + \beta_1) \tag{2.25}$$

$$K(\vec{x_i}, \vec{x_j}) = e^{\frac{-|\vec{x_i} - \vec{x_j}|^2}{2\sigma^2}} \tag{2.26}$$

All we have to do in order to apply SVMs to non-linear problems is to replace everywhere we see $\vec{x_i} \cdot \vec{x_j}$ with some suitable kernel function $K(\vec{x_i}, \vec{x_j})$. The function given in (2.26) will be the one we employ. It's known as the *Gaussian kernel*, and has been shown to give good performance over a variety of different problems. It has one free parameter, σ, which controls the width of the Gaussian function. Its value should be chosen to be of roughly the same magnitude as the length of your training vectors. The actual value should be determined through an empirical study of different trainings on your specific problem in order to optimize the SVM's performance. As a general rule, σ determines how flexible the learned function's contours can become in the problem's unmapped feature space. The larger the magnitude of σ, the more rigid (less bendy) the contours. In fact, the Gaussian kernel function tends toward the linear kernel (*i.e.* Equation (2.23)) as σ tends toward ∞. If σ is taken to be too small, the SVM will simply "memorize" your training set and not generalize well.

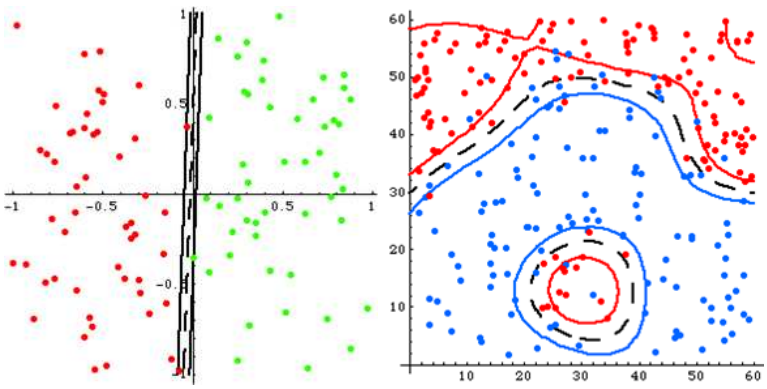


Figure 2.4: Linear and non-linear problems

2.5 Feature Ranking & Reduction

Another interesting property of the SVM formalism is that the learned function describes a geometrical space which, in and of itself, can be analyzed. This provides additional information above and beyond the simple binary answers derived from the learned function's sign. One piece of information which can be very insightful in analyzing a classification problem is a ranking of the features from most to least relevant.

In a SVM, each of the features is represented by one of the dimensions in the space defined by the learned function. The most straight forward way to achieve this ranking is by taking the gradient of the learned function (in its d dimensions). By then sampling the gradient at different points, a numerical ordering of the most relevant features can be found. Recall, the learned function is defined as

$$f(\vec{z}) = \sum_{i=1}^{N} \alpha_i y_i K(\vec{x_i}, \vec{z}) - b \tag{2.27}$$

For a Gaussian kernel function, this becomes

$$f(\vec{z}) = \sum_{i=1}^{N} \alpha_i y_i e^{\frac{-|\vec{x_i} - \vec{z}|^2}{2\sigma^2}} - b \tag{2.28}$$

Note that the sum index i in this context runs over the N support vectors. Thus $\vec{x}_i$ refers to the i^{th} support vector, and not the i^{th} dimension of $\vec{x}$. The j^{th} component of the gradient $(\vec{\nabla} f)_j$ is then given by

$$(\vec{\nabla} f)_j = \frac{\partial}{\partial z_j} f$$

$$\sum_{i=1}^{N} \alpha_i y_i e^{\frac{-|\vec{x_i} - \vec{z}|^2}{2\sigma^2}} = \sum_{i=1}^{N} \alpha_i y_i e^{\frac{-(\vec{x_i}_1 - z_1)^2 - (\vec{x_i}_2 - z_2)^2 - \ldots - (\vec{x_i}_j - z_j)^2 - \ldots - (\vec{x_i}_d - z_d)^2}{2\sigma^2}}$$

$$\Rightarrow (\vec{\nabla} f)_j = \sum_{i=1}^{N} \frac{\alpha_i y_i}{\sigma^2} (\vec{x_{ij}} - z_j) e^{\frac{-|\vec{x_i} - \vec{z}|^2}{2\sigma^2}} \tag{2.29}$$

where by $\vec{x_{ij}}$ it is meant the j^{th} component of the i^{th} support vector.

For completeness, the gradient for a linear kernel is

$$f(\vec{z}) = \sum_{i=1}^{N} \alpha_i y_i \vec{x_i} \cdot \vec{z} - b$$

$$(\vec{\nabla} f)_j = \sum_{i=1}^{N} \alpha_i y_i \vec{x}_{ij} \tag{2.30}$$

To see why this method works, consider a simple two dimensional problem and a linear SVM. The decision plane and the margin define a surface of constant slope and orientation. The gradient in this situation is then a constant vector in some direction which is normal to the decision plane. Now, if one of the dimensions (features) turns out to be completely irrelevant to our decision, then our problem was really one dimensional. In this case we would find a decision plane parallel to the ignorable dimension, and a gradient pointing along the dimension that matters. Extended over many dimensions, the same general behavior still holds. The gradient points in the direction of the most relevant features, and is weighted in proportion to the pertinence each feature has with respect to the decision. See Figure 2.5 below.

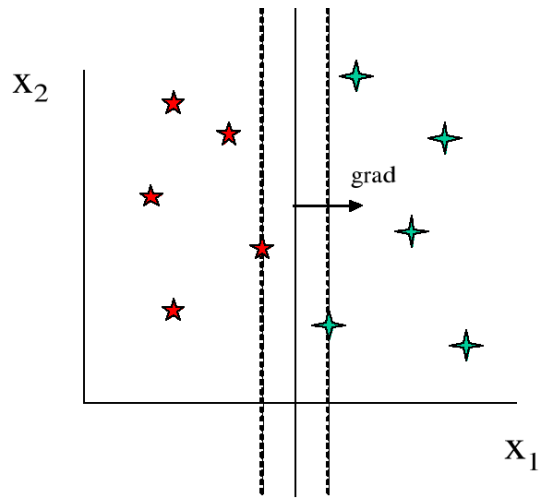


Figure 2.5: Using the gradient to rank features

In the non-linear case, the gradient varies from point to point in the decision space. To cope with this situation, we take the average of the gradient for some appropriate number of test points in the area of interest. This may have some drawbacks, as the averaging might wash out over a feature that is positively and negatively correlated over the test sample in different regimes. However, as an intuitive tool, the gradient is

a very nice way to gain confidence in your classifier's behavior and to better understand the dataset.

2.6 Implementation

To create a functional Support Vector Machine, it is required to solve the quadratic optimization problem given in Equation (2.21) with the appropriate kernel function included. The SVMs used in this thesis were implemented by the author and Jacob Borgman in C++. The implementation we used is based upon the improvements to Platt's Sequential Minimal Optimization (SMO) [30] method proposed by Keerthi, *et al.* [29]. This implementation was found to be faster than general quadratic optimization software, and other existing SVM implementations.

Chapter 3

Multiclass SVM

The previous sections have described how to use SVMs to solve binary classification problems. Many problems can be cast in this form, and be satisfactorily solved using a single SVM. Some unique properties of the physics problem we are attempting to solve in this thesis have suggested the use of a multiclass SVM system. Because the fractional heavy quark content of the W+Jets backgrounds in our problem are not well known, the results given by a single SVM depend greatly on the amount of heavy quarks used to compose the backgrounds in our training sets. In order to avoid having to pick a specific mixture of heavy *vs.* light quarks in the W+Jets backgrounds, it was decided to employ a multiclass SVM framework that treats the W+heavy quark flavors as a separate background from W+light flavors. This allows the heavy flavor fraction of the background to be measured independently of the $t\bar{t}$ signal, and therefore removes the bias caused by having to pick a particular training mixture for the two-class problem. These next sections explain how to use several SVMs to attack a general multiclass problem. Then, some specifics of the three-class problem which was used in computing the $t\bar{t}$ cross section are provided.

3.1 Generalized Solution

Suppose there are many objects, belonging to N different classes, that are to be separated using SVMs. Instead of the single SVM of the $N = 2$ case, we will now use a system of multiple SVMs to accomplish classification. After the creation of training sets representative of each data class, a SVM is trained for each pairing of classes in our problem. This leads to the creation of $K = N(N-1)/2$ trained machines. These

SVMs then form a set of classifiers to which one may take an event of unknown class, and systematically ask each SVM the following question: "Assuming this event belongs to either your $y = +1$ class or your $y = -1$ class, to which does it belong?" How one goes about answering this set of questions can be different depending on the application at hand. For example, one could simply examine the sign of the learned function for each of the SVMs. If simple classification is the only requirement, then a popular method is to count the number of votes each class generates. However, for our purposes, we would like something a little bit more rigorous and quantitative. It would be much preferable to work in terms of an estimated class probability for each event. The method of *Pairwise Coupling* [3] provides a means to make this kind of estimate. Also, it was found that a direct examination of a combined feature space for the trained SVMs can be used to compute these probabilities. Both of these methods are described below.

3.2 Pairwise Coupling

Once our system of K Support Vector Machines has been trained, we can assemble a matrix of pairwise probabilities for any proposed test vector $\vec{z}$. This matrix $r_{ij} = \text{Prob}(\vec{z} \in i \mid \vec{z} \in i \text{ or } \vec{z} \in j)$, where i and j run over the N classes. Clearly, the diagonal is undefined ($i \neq j$), and $r_{ij} = 1 - r_{ji}$. We would like to discover the class probabilities $p_i = \text{Prob}(\ \vec{z} \in i\ )$, such that $\sum_i^N p_i = 1$, which are compatible with the pairwise probabilities r_{ij}.

The Pairwise Coupling procedure uses the model that $r_{ij} = p_i/(p_i + p_j)$. It assumes this form for r_{ij} and computes the p_i's so that a distance between a $\mu_{ij} \equiv p_i/(p_i + p_j)$ and r_{ij} is extremized. The metric employed by Hastie & Tibshirani in their paper is the Kullback-Leibler distance (also known as the relative entropy in physics!). This distance is specified as

$$\ell(\vec{p}) = \sum_{j=1}^{N} \sum_{i<j} [r_{ij} \log \frac{r_{ij}}{\mu_{ij}} + (1 - r_{ij}) \log \frac{1 - r_{ij}}{1 - \mu_{ij}}] \tag{3.1}$$

where $\vec{p} \equiv \{p_1, \ldots, p_N\}$, and is re-expressed as μ_{ij} on the right hand side.

The r_{ij} matrix is created by taking the value of the learned function $f(\vec{z})$ [see Equation (2.27)] from the SVM trained on classes i and j. This value is then used to evaluate that SVM's probability distribution functions (PDFs). There are $2K$ PDF

functions, as each class pairing results in a SVM, and each SVM has a PDF for its $y = +1$ class and its $y = -1$ class.

The class i PDF for the (i *vs.* j) trained SVM is taken to be PDF_i^{ij}. Then, we form $r_{ij}(\vec{z}) = \frac{PDF_i^{ij}(\vec{z})}{PDF_i^{ij}(\vec{z})+PDF_j^{ij}(\vec{z})}$. It will be seen later that these PDFs are found empirically for our problem, but for the purposes of this discussion can be assumed to be supplied by an oracle.

Hence, finding the $\vec{p}$ that maximizes $\ell(\vec{p})$ provides an estimate of the class probabilities for an unknown event $\vec{z}$. The Pairwise Coupling scheme outlined here was implemented in C++ for use in this analysis.

3.3 Feature Space

A more direct way of determining the separation ability of the SVM system is to view the problem's feature space and how the different data classes are distributed therein. Because we take our problem to consist of three classes, we find ourselves with a system of $3(3-1)/2 = 3$ SVMs. As is seen in Equation (2.2), each SVM has a normal vector in feature space given by $\vec{w}$. These vectors can be written as

$$|w_1\rangle = \sum_i^{SVM1} y_i \alpha_i |\varphi(\vec{x}_i)\rangle \tag{3.2}$$

$$|w_2\rangle = \sum_i^{SVM2} y_i \alpha_i |\varphi(\vec{x}_i)\rangle \tag{3.3}$$

$$|w_3\rangle = \sum_i^{SVM3} y_i \alpha_i |\varphi(\vec{x}_i)\rangle \tag{3.4}$$

where φ is the unknown mapping function that takes our initial vectors into the feature space used in constructing our separating hyper-plane. Here we are assuming that each of the 3 SVMs in question have been trained using the same kernel function and hyper-parameters. Generally, the dimensionality of the mapped space is very large. At the same time, the way in which these $|w\rangle$ vectors intersect is almost surely non-orthogonal. Because this is true, the simple plotting of a 3-tuple for a test point $\vec{z}$ given by $(f_1(\vec{z}), f_2(\vec{z}), f_3(\vec{z}))$ can misrepresent the actual separation of points in feature space. The value of the learned function $f(\vec{z})$ simply offers a measure of the distance between a test point and that SVM's hyper-plane. The relative orientations

of the 3 SVMs in feature space is not taken into account. What we would really like is to construct an ortho-normal basis in feature space using the 3 $|w\rangle$ vectors. This can be done using the Gram-Schmidt procedure as follows:

$$|w_1'\rangle = |w_1\rangle \tag{3.5}$$

$$|w_2'\rangle = |w_2\rangle - \frac{\langle w_1|w_2\rangle}{\langle w_1|w_1\rangle}|w_1\rangle \tag{3.6}$$

$$|w_3'\rangle = |w_3\rangle - \frac{\langle w_1|w_3\rangle}{\langle w_1|w_1\rangle}|w_1\rangle - \frac{\langle w_2'|w_3\rangle}{\langle w_2'|w_2'\rangle}|w_2'\rangle \tag{3.7}$$

Using this basis, we can then create 3-tuples (X, Y, Z) like this:

$$X(\vec{z}) = \frac{\langle \vec{z}|w_1'\rangle}{\langle w_1'|w_1'\rangle} \tag{3.8}$$

$$Y(\vec{z}) = \frac{\langle \vec{z}|w_2'\rangle}{\langle w_2'|w_2'\rangle} \tag{3.9}$$

$$Z(\vec{z}) = \frac{\langle \vec{z}|w_3'\rangle}{\langle w_3'|w_3'\rangle} \tag{3.10}$$

See Appendix A for an expansion of these equations. This expansion can be used to create a function that takes a 3-tuple of the learned function values, and returns a 3D point from feature space as above. Using such a function, I have plotted the zero contour of the learned function for each of the three SVMs used in this physics analysis. This maps out the hyper-plane decision surfaces in feature space, and illustrates the non-orthogonality described above. See Figures 3.1 and 3.2. These figures also show points sampled from each of the three major classes of training data, and therefore illustrate their separability. Note that this three dimensional (3D) plot can be viewed interactively by visiting `http://tuhept.phy.tufts.edu/~ben/talk1.html`.

3.4 Summary

As was seen above, SVMs can be used effectively to tackle multi-class problems. There are a few considerations in choosing the best method for extracting the multi-dimensional information from the SVM systems. While Hastie's Pairwise Coupling is

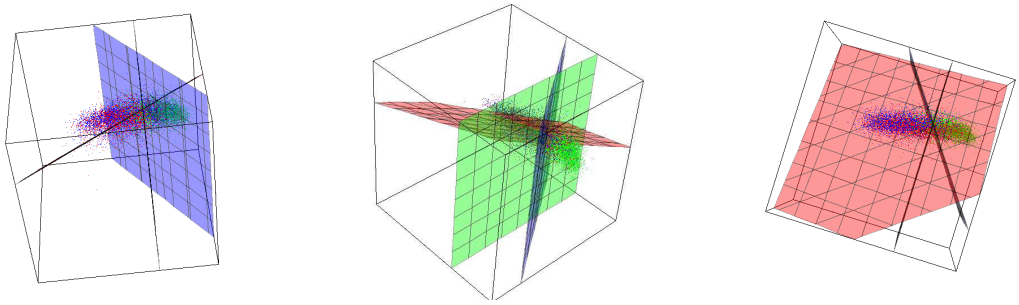


Figure 3.1: Kinematic SVM's ortho-normalized data points, plotted in feature space.

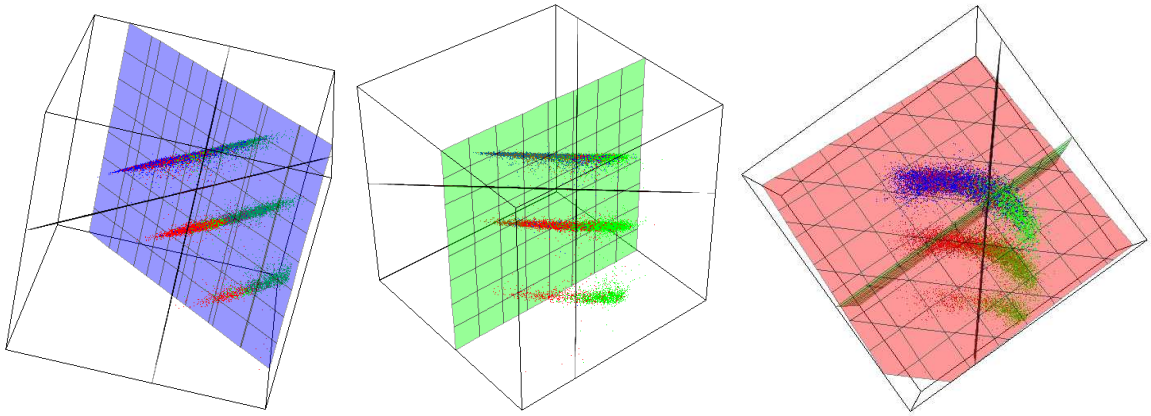


Figure 3.2: b-tagging SVM's ortho-normalized data points, plotted in feature space.

straight forward and convenient, it was found to introduce a small bias in the pseudo-experiments which employed it. The feature space method was found to be largely un-biased in pseudo-experiments, and as such it was used as the primary method of producing probability (density) estimates. Pairwise Coupling was then used mainly as a cross check on our methodology in the final $t\bar{t}$ physics analysis.

Part III

Apparatus

Chapter 4

Fermilab and the Tevatron

The purpose of this chapter is to give an overall picture of the Fermi National Accelerator Laboratory (FNAL), and specifically outline the operation of the particle accelerator & collider located there. FNAL is run under the control of the United States Department of Energy, and was built in 1967 for $243 million. The Tevatron itself was completed in 1983 for a mere $120 million, though it used much of the laboratory's existing infrastructure to subsidize the project. Fermilab employs roughly 2,000 people, and its total operating budget for FY2008 was $320 million. It also serves as host to a large community of visiting scientists and students from all around the world. See [18] for more information.

Figure 4.1: Wilson Hall, and the Tevatron

4.1 Tevatron

The Tevatron is located in suburban Illinois, about 40 miles west of Chicago, at the Fermi National Accelerator Laboratory (FNAL). It is a proton & anti-proton collider ($p\bar{p}$), is 2π kilometers in circumference (6.28 km $\approx$ 3.91 miles), and has two main interaction points which are approximately separated by 120°. The measurement described in this thesis was carried out at one of these locations using the CDF (Collider Detector at Fermilab) detector. This chapter examines the various pieces of the Tevatron in general, and the next chapter concentrates on the specifics of the CDF detector[16].

The first task required to accelerate protons and anti-protons is, of course, to gather a supply of these objects. Fortunately, protons are easy to come by in our area, and the protons used at the Tevatron come from your standard common tank of compressed hydrogen. On the other hand, anti-protons are quite scarce in our world, and hence they will need to be manufactured in our accelerator chain. Like the gears of an automobile, there are certain types of accelerators which work better for different particle energy regimes. From initial energies near 25 keV, all the way up to the targeted 0.98 TeV per beam, the Tevatron uses a series of four (4) machines in assistance: Pre-accelerator, Linac, Booster, and the Main Injector.

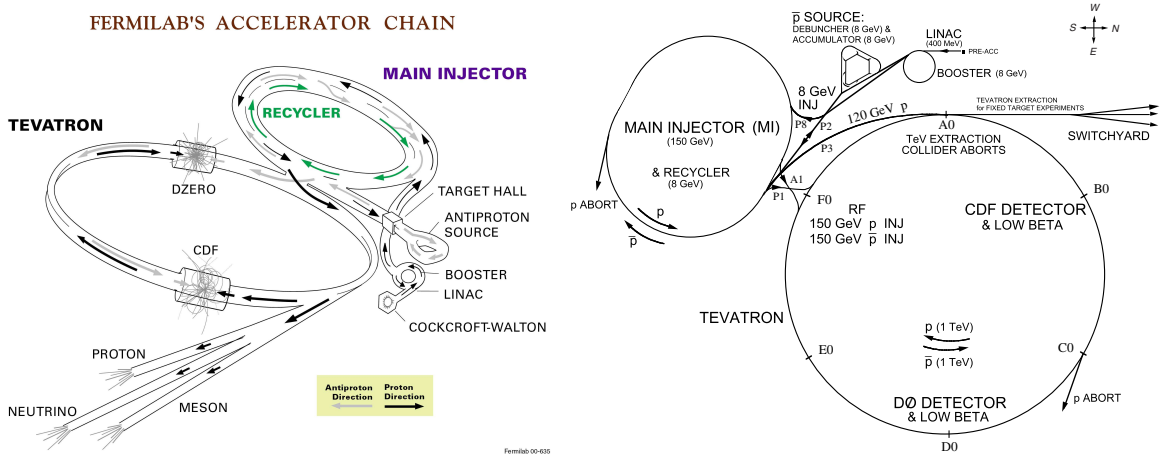


Figure 4.2: Acceleration Chain

4.1.1 Cockroft-Walton

To start the acceleration chain, a Cockroft-Walton style accelerator is used to bring negative hydrogen ions (H^-) to an energy of 750 keV. In particle physics, the electron

volt (eV) is often the preferred unit of energy, and is of course the amount of energy imparted to a single electron when accelerated through an electrostatic potential of one (1) volt. Richard Feynman summarized the situation very nicely in a lecture at Caltech in the early 1960s[17]: "...a single atom is such a small thing that to talk about its energy in joules would be inconvenient. But instead of taking a definite unit in the same system (like 10^{-20} J), [physicists] have unfortunately chosen, arbitrarily, a funny unit called an electron volt, ... and that turns out to be about 1.6×10^{-19} J. I am sorry that we do that, but that's the way it is for the physicists."The hydrogen from the storage tank is ionized (in this case, given an extra electron), and then kept in a dome which is maintained at an electric potential of -750,000 volts. The negative ions accelerate through this potential to the grounded pre-accelerator wall. A transport line then guides the accelerated ions to the Linac[12].

Figure 4.3: Cockroft-Walton and the Linac

4.1.2 Linac & Booster

The "Linac", short for linear accelerator, is next in the acceleration chain. This machine uses a series of 14 radio-frequency cavities to push up the energies of the ions further to 400 MeV[13]. After leaving the Linac, the ions are passed on into the Booster. The first thing done at the Booster stage is to strip all the electrons off the H^- ions, leaving only the proton nuclei. This is done by passing the ions through a thin foil of carbon. The Booster is a synchrotron (circular accelerator) that extends 150 meters in diameter, and is made up of a series of magnets with 18 RF cavities interspersed. The magnets are used to bend the beam into its circular path, and short areas of applied electromagnetic potential increase their energy. At injection, the magnetic field is 740

gauss, and by extraction it has reached 7,000 gauss. The process of accelerating the protons in this ring tends to cause them to coalesce in stable regions of acceleration called buckets. These buckets give rise to groupings of protons, or *bunches*, which typically contain 6×10^{10} protons each. At the end of the Booster phase, the protons have reached 8 GeV, and are ready to be passed along into the Main Injector.

4.1.3 Main Injector

The Main Injector is a much larger synchrotron. It is seven (7) times larger then the Booster, and is able to accelerate the 8 GeV protons to an energy of first 120 GeV, and then 150 GeV. The smaller bunches from the Booster are regrouped at this stage into larger collections of 6×10^{13} protons. Thirty-six (36) of these bunches are created, and injected into the Tevatron separated by 396 nanoseconds.

It is at this stage in the acceleration chain that the anti-protons are produced. Every 1.5 seconds, some of the 120 GeV protons are directed at a target composed of nickel (Ni). The results of this fixed target interaction produces anti-protons in the debris. These anti-protons are focused via a lithium lens, and captured using a magnetic field. Upon leaving the target, the anti-protons have a somewhat wide energy spectrum in the neighborhood of 8 GeV. For every million (10^6) protons sent onto the nickel target, only approximately 20 anti-protons survive to be used for collisions in the Tevatron.

Because the protons arriving on target are bunched, the resulting $\bar{p}$'s are also bunched. In order to be acceptable to subsequent higher energy accelerators in the chain, it is desirable that the $\bar{p}$'s are rotated in phase space to produce a more uniform momentum and continuous beam. This is done through a stochastic cooling process in an 8 GeV synchrotron called the Debuncher. Once this process is realized, the $\bar{p}$'s are transfered to another 8 GeV synchrotron called the Accumulator. Its purpose is to take groups of 8 GeV anti-protons from the Debuncher and create "stacks" of roughly 10^{11} particles. These 10^{11} $\bar{p}$'s are used to create four bunches in the Tevatron. When there are enough anti-protons in the Accumulator to complete a "store", they are transfered in groups back into the Main Injector where they are accelerated up to 150 GeV and passed on to the Tevatron. Thirty-six (36) bunches of anti-protons are injected in this fashion, in opposition to the protons already present in the machine.

4.1.4 Tevatron Main Ring

The final stage of acceleration is done in the main Tevatron ring. It is a synchrotron made up of some thousand (1,000) superconducting magnets. To maintain their superconducting state, the magnets are cooled to 4 K. They produce a magnetic field of 4.2 telsa (42,000 gauss) which is used to keep the beam within its almost 4 mile circular orbit. The protons rotate clockwise (looking from above), and the anti-protons counter-clockwise. Each beam is only about the width of a human hair ($\sim$30 μm), and traces out a helical path which intertwines its antimatter partner in a caducean fashion. At their peak energy, each beam reaches an energy of 0.98 TeV, which allows for a center of mass collision energy $\sqrt{s} = 1.96$ TeV.

To put this energy into perspective, the Tevatron's proton beam contains about 2×10^{15} protons. This means that there is roughly 35 million joules of energy circulating in the ring once the protons have reached 1 TeV. That amount of energy is approximately the same as would be required to lift an elephant (8 tons, 7,250 kg) to the top of the Sears Tower (442 meters) in Chicago. If the elephant were to fall off the building, the amount of energy it would have upon reaching the ground is the same as is stored in the Tevatron during normal operation!

The Tevatron ring is sectioned into 60 degree segments labeled A through F. Each of these segments are sub-divided into five (5) sections which are numbered 0 through 4. The zero sections are actually short straight-away segments for the beams, and hence it's in these sections the two major interaction points and collision detectors for each of their experiments are located. The Collider Detector at Fermilab (CDF) experiment is located at the B0 section in the ring, while the DØ (pronounced "dee-zero") experiment is aptly housed in section D0. Proton / anti-proton collisions are induced through the use of "low beta quadrapole" magnets, which act to squeeze the two beams together at the desired interaction point. The typical instantaneous luminosity for CDF at the beginning of a store is 9.5×10^{31} cm^{-2} s^{-1}. Since the beginning of Run II at the Tevatron in 2001, the total time integrated luminosity has reached in excess of 3 fb^{-1}.

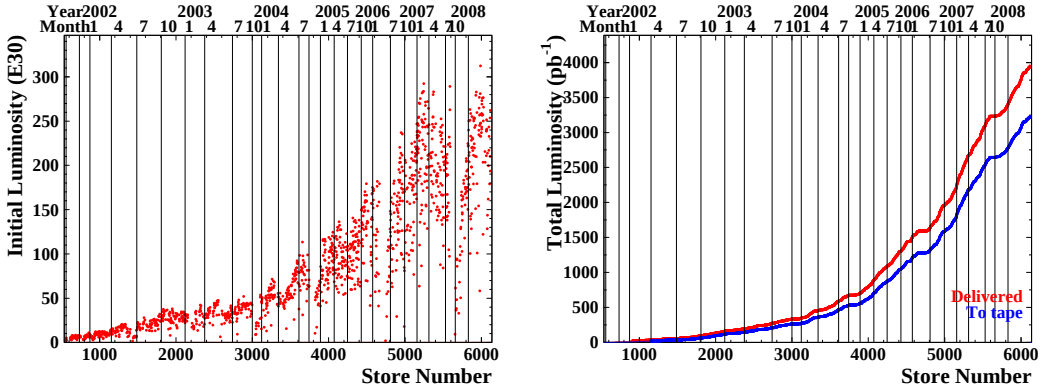


Figure 4.4: CDF Luminosity by Store

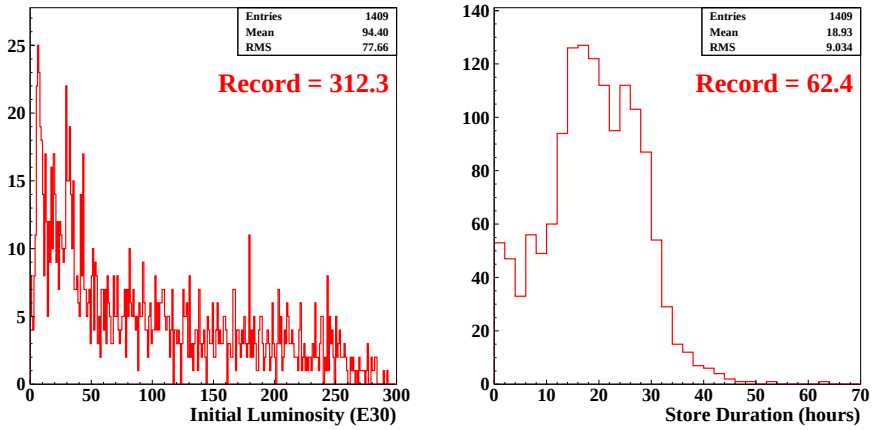
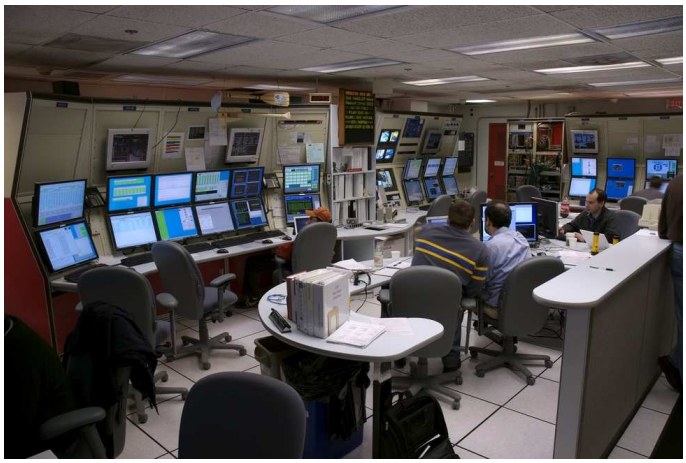
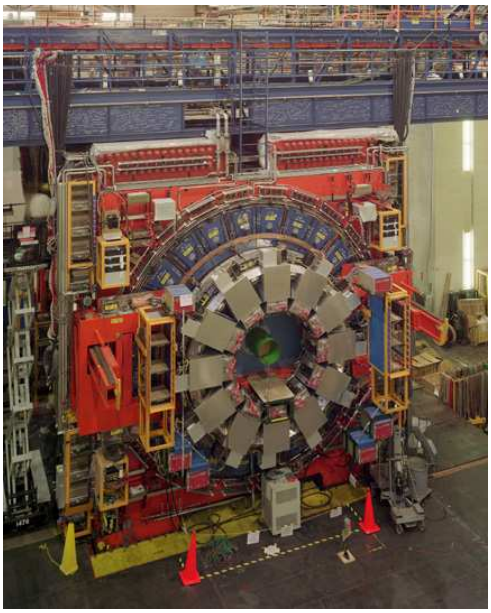


Figure 4.5: CDF Store Profiles

Chapter 5

Collider Detector at Fermilab





The CDF detector is based around a solenoid, which is then surrounded by calorimetry, and muon detection systems. Inside the solenoid, there are charged particle tracking systems, consisting (from the inside-out) of a silicon vertex detector abutting the beam pipe, and an open cell drift chamber called the Central Outer Tracker (COT). Located at the B0 hub of the Tevatron ring, the detector sits centered 10 meters underground. The protons in the ring move clockwise from above, and enter the detector moving West to East. Anti-protons circle in the opposite direction and enter the detector from the East side. The coordinate system employed at CDF usually puts the origin at the interaction point in the center of the apparatus. The z axis extends into

the proton direction (East) along the beamline, while the x and y axes create a right handed coordinate system such that the x-axis is in the same plane as the Tevatron ring. Hence, the polar angle θ is measured from the West, the azimuthal angle $\phi = 0°$ looks North, and $\phi = 90°$ points up into outer space. Also, it is customary to use a defined quantity called pseudorapidity[1] in lieu of the polar angle θ, where $\eta \equiv -\ln[\tan(\theta/2)]$. Many quantities in hadron colliders are "flatter" as functions of η as opposed to the polar angle θ. Pseudorapidity is convenient as it takes on a value of zero at $\theta = 90°$, and then varies symmetrically from ∞ to $-\infty$ as θ runs over its domain from 0° to 180°. However, it doesn't take on very large values (greater then 10) until θ becomes a very "forward" angle, which is to say that the angle θ is so small or large that it points down the beamline. See Figure 5.1.

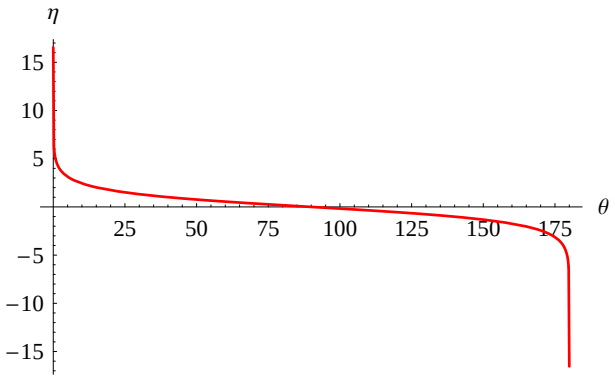


Figure 5.1: Pseudorapidity $\eta(\theta)$

5.1 Tracking

5.1.1 The Silicon

One of the fundamental problems in extrapolating tracks from the secondary particles created in the $p\bar{p}$ collisions is finding their point of origin. Without detectors close up to the interaction region, it becomes very hard to differentiate between hard tracks that come from the primary interaction, and the tracks from the decay of other particles with very short lifetimes which create a secondary vertex which is slightly displaced

[1]This quantity is often sloppily called rapidity, though it is really only the massless approximation of the actual Lorentz additive quantity $y = \frac{1}{2}\ln\frac{E+p_z}{E-p_z}$ bearing that name.

from the main collision point. For instance, B hadrons can decay within a few hundred picoseconds of their creation, and only travel several millimeters before turning into longer-lived particles visible to the rest of the detector. Without eyes up front, pinpointing the exact locus for certain tracks to within millimeters would be impossible. With this in mind, the innermost tracking system of the CDF detector is tasked to provide data points for the paths we reconstruct as close as possible to the beam line.

This initial tracking instrument is composed of three different detector systems which are collectively and colloquially called "the silicon". These are the "Layer 00"(L00), the Silicon Vertex Detector (SVX II), and the Intermediate Silicon Layers (ISL).

Layer 00 resides at a radius from the beamline of 1.35 centimeters. L00 is a single sided strip sensor, and is attached right upon the beam pipe. It is composed of 12 flat strips which are arranged around the beamline. This configuration gives full coverage in ϕ, and a length along the z-axis of $\pm$475 mm [19].

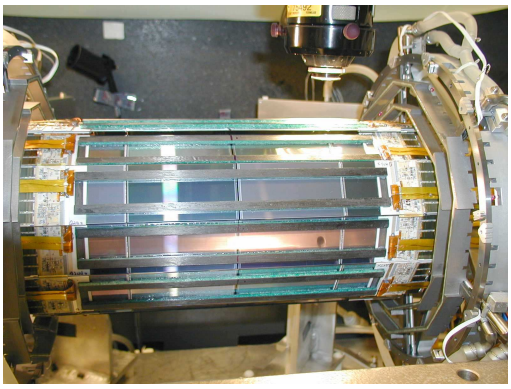


Figure 5.2: SVX II under construction

Five concentric detector layers make up what is called the Silicon Vertex Detector, which is abbreviated as SVX II in order to differentiate it from its CDF Run I incarnations. Three identical barrels of length 29 centimeters each are stacked end-on-end. This gives a z-axis extent of $\pm$435 mm. Each barrel is subdivided into 12 wedges in ϕ. Each wedge is configured as a "ladder" consisting of five rungs of double sided silicon strip sensors. The first resides at a radius of 2.45 cm, and the last extends out to 10.7 cm. In the first three layers, one side of the silicon strip gives an axial r-ϕ position, while the other side reports a stereo measurement oriented at 90° to its partner. The final two rungs on the ladder also give an axial r-ϕ measurement, but in stereo with

the strip information shifted by 1.2°.

The ISL is the outermost tracking system using silicon, and provides one last set of data points to particle tracks before they enter the Central Outer Tracker (COT). As can be seen in Figure 5.3, the COT only provides its full radial coverage out to an $|\eta| < 1$. As can also be seen in the figure, the SVX II is designed to provide tracking information even out to a forward region with $|\eta| < 2$. For $|\eta| < 1$, the combination of the SVX II and the COT could provide full three-dimensional (3D) tracking, but the bulk of the tracking information would come from the COT, and hence the overall resolution suffers. In the forward region $1 < |\eta| < 2$, the problem is even worse as the COT coverage diminishes quickly with η. The ISL was designed with these problems in mind, and serves to "glue together" the SVX II with the COT. In the central region an extra double-sided silicon strip is placed at a radius of 22 cm. The large η regions of the detector are given two new layers of information, placed at 20 cm and 28 cm. These extra layers augment the SVX II/COT, and provide enough information to allow for complete 3D tracking for all $|\eta| < 2$.

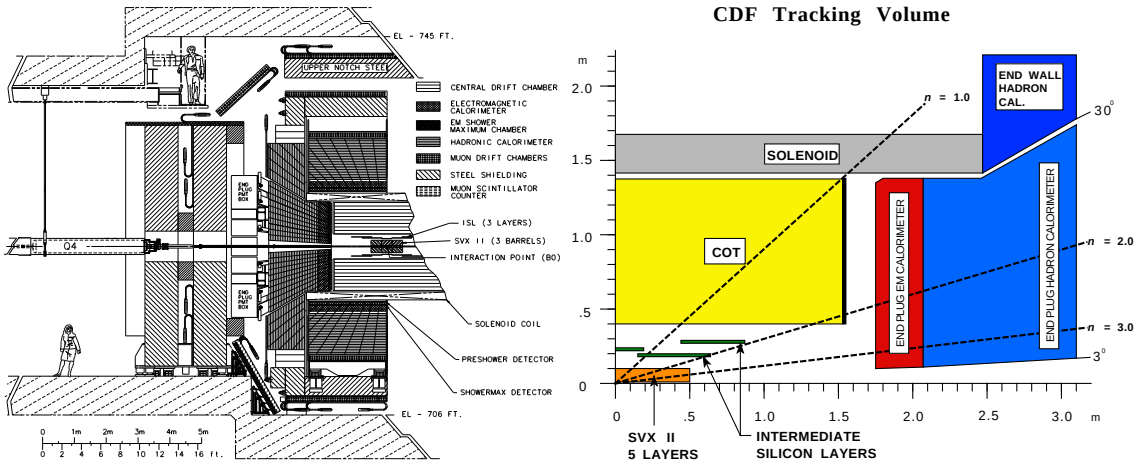


Figure 5.3: View of CDF

5.1.2 Central Outer Tracker

The Central Outer Tracker (COT) is an open cell drift chamber used to reconstruct charged particle tracks at CDF. It is located within the CDF solenoid just outside the ISL, and extends in the z direction ± 155 cm for an overall length of 310 cm. Its cylindrical volume starts at a radius from the beamline of 43 cm, and extends out to

133 cm. The general principle used to locate tracks within this active volume is as follows. Several layers of wires, each extending the whole length of the detector along the z direction, are placed into groupings called cells. The wires are fixed at a high voltage with respect to the cell's wall. The whole of the detector is filled with a specially chosen gas mixture, which when traversed by high energy particles, causes the gas to be ionized. The ions and electrons created through this process are accelerated by the potential toward the wires where charge collects and an electric current is produced. By recording which wires in the layers read out a current, it is possible to see the particle paths through the COT. Through a careful analysis of the amount of current induced and how long it took for the current to read out, it is possible to calculate just how close to the wires a particle actually passed. In this way, the track finding resolution is increased well beyond the simple coarse spacing of the wires themselves.

The COT for Run II at CDF is composed of 8 super-layers, which extend out in the radial direction. Each super-layer is composed of cells that each have 12 sense wire layers. This means that for a track which traverses the whole of the COT, a total of 96 (8×12) separate position measurements are available for track reconstruction. There are also 13 field wires within each cell which are used to control the shape of its electric field. The sense wires are held at a voltage of 2,000 volts, while the potential wires are run at a higher voltage around 3,000 volts. The actual voltage of each potential wire is varied somewhat to create a uniform drift field within each cell. For tracks with an $|\eta| < 1$, all 8 super-layers are exposed to the trajectory. However, tracks with $|\eta| > 1$ will start to lose information. Once past $|\eta| > 1.3$, only fewer then half (≤ 3) of the super-layers are traversed.

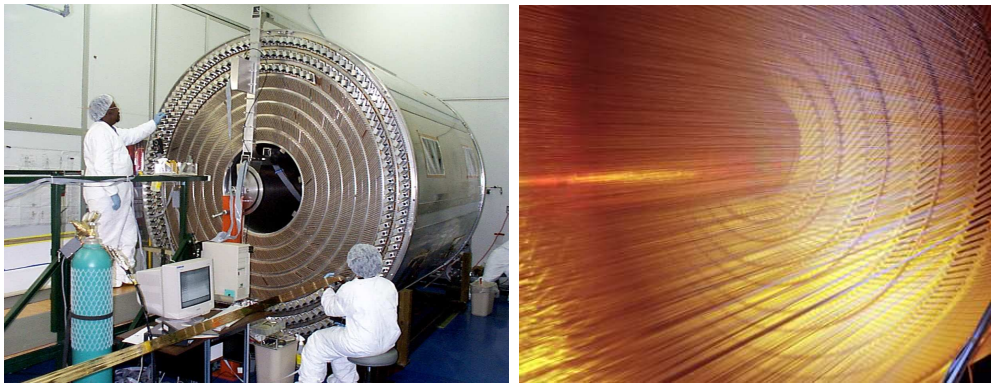


Figure 5.4: Inside the COT

The super-layers alternate between stereo and axial measurements, with an offset of $\pm 3^\circ$ for the stereo layers. This simply means that while the axial layers run parallel to the z-axis and give r-ϕ information, the stereo super-layers are placed at a small angle with respect to z, and the resulting grid formed by the overlapping super-layers yields the z position for a hit. All told, there are a total of 30,240 sense wires and 32,760 potential wires within the chamber. In order to take into account the effect of the Lorentz force of the CDF solenoid's 1.4 tesla magnetic field on particle trajectories, the COT's cells are tilted at an angle of 35°.

The gas used to fill the chamber is chosen so that the drift time for (velocity of) the ions created will be quick enough to be read out before the next beam crossing. To achieve this at CDF, a gas mixture of Argon-Ethane-CF_4 in proportion (50:45:5) is used. An Argon-Ethane (50:50) mixture has been shown to result in a maximum drift time of 100 ns, which is ample for the Tevatron's 396 ns time between bunch crossings. The small amount of added CF_4 is used to reduce aging effects on the wires[21].

See Figure 5.6 for summary information concerning the performance characteristics of all the tracking systems at CDF.

Number of Layers	96
Number of SLs	8
Sense Wire Spacing	7.62 mm
Wire Diameter	40 μm gold plated tungsten
Wire Tension	135 g
Drift Field	1.9-2.5 kV/cm
Maximum Drift Distance	0.88 cm
Maximum Drift Time	100 ns
Tilt Angle	35°
PseudoRapidity Coverage (All SLs)	$\lvert\eta\rvert < 1$
Length of Active Region	310 cm
Total Number of Wires	73,080
Endplate Load	40 metric tons

Super-layer	1	2	3	4	5	6	7	8
Stereo Angle	$+3^\circ$	0°	-3°	0°	$+3^\circ$	0°	-3°	0°
Cells/Layer	168	192	240	288	336	384	432	480
Sense Wires/Cell	12	12	12	12	12	12	12	12
Radius at SL Center (cm)	47	59	70	82	94	106	117	129

Figure 5.5: COT Design Summary

Detector(s)	Feature	Resolution
L00	hit position	11 μm
SVX	hit position	5 μm
SVX+ISL	z_0 position	40 μm
COT	hit position	140 μm
	momentum $\frac{\sigma(p_T)}{p_T}$	0.15% $p_T[GeV/c]^{-1}$
SVX+ISL+COT	momentum $\frac{\sigma(p_T)}{p_T}$	0.07% $p_T[GeV/c]^{-1}$

Figure 5.6: Tracking Summary

5.2 Time of Flight

Just outside the COT, but still within the CDF solenoid, resides a detector of radial thickness 4.2 centimeters called the Time-of-Flight system (TOF)[22]. It is made up of 216 scintillator bars for full coverage in ϕ, and whose function is to measure the time duration for a particle from the main interaction region to reach them. Located at a radius of 1.4 meters from the beamline, this time is typically on the order of 5 nanoseconds for the fastest particles. Hence, the TOF has a high resolution of 100 picoseconds. It has two major purposes in particle detection and identification. First, it plays a large role in the cosmic ray veto algorithm used to identify tracks that do not originate from the colliding particles in the Tevatron. Second, it is used to discriminate between stable particles of different mass that have made their way through the tracking volume.

$$m = \frac{p}{c}\sqrt{\frac{t^2c^2}{L^2} - 1} \tag{5.1}$$

The mass m of the particle can be reconstructed through the use of Equation (5.1), where p is the momentum from the CDF tracking systems, t the flight time, and L the path length. The ability to distinguish charged kaons and pions aids in tagging the decays of particles containing b quarks, which is of central importance to most bottom and top physics analyses at CDF.

5.3 Calorimetry

Once outside the tracking systems, we would like to then measure the energy of the particles emanating from the collision. The tracking information up to this point

gives us the 3-momentum (p_{xyz}) through the examination of track curvature through the CDF magnetic field. The Lorentz force from this field is non-perturbative to the particles, in the sense that their deflection through the field doesn't change their initial momentum's magnitude. Since the real feature of interest is the particle 4-momentum (E, p_{xyz}), we must be able to measure p without disturbing $|p|$ and E very much. Thus, it is not by coincidence that the detectors used up to this point in the reconstruction process contain as little "stuff" as possible. Thin layers of silicon and wire and their support structures are all that have been in the way of the flying debris from the $p\bar{p}$ interaction. In order to finally measure the particle energies, things become a bit more invasive. In fact, the major goal of the calorimetry put in place outside the solenoid is to stop all possible particles of reasonable energy, while in the process of measuring that energy.

The calorimetry at CDF is separated into two major regions which are mirrored on the East and West sides of the detector. The central region covers $|\eta| < 1.1$, while the forward regions extend to $1.1 < |\eta| < 3.6$. Each region in η is further divided into two type of sensors which distinguish between energy deposited by electrons and photons (called Electromagnetic or EM), and energy from hadronic jets. The central systems are called the Central ElectroMagnetic and Central HAdronic (CEM and CHA) calorimeters. The large η, or "plug", systems are the Plug ElectroMagnetic and Plug HAdronic (PEM and PHA). There is also a set of hadronic calorimeters known as the WHA (End Wall Hadronic) that fills a gap between the CHA and PHA. See Figure 5.3 for an illustration of the configuration.

All the calorimeters used at CDF operate on the principle of stacking layers of scintillator and a heavy absorbing material. The absorber causes the incoming particle to shower, and then the size of the shower is sampled through the use of a scintillator. The scintillator produces photons in response to the shower which are "read out" through photo-multiplier tubes (PMTs). The number of photons read out is then proportional to the number of particles in the shower, which is in turn proportional to the energy of the incident particle.

In the electromagnetic case, the absorber used is lead (Pb), and incoming $e^{\pm}$'s and photons (γ's) of sufficient energy cause a shower to form. Electrons and positrons undergo bremsstrahlung (produce photons), while photons cause pair production (create electrons and positrons). As one might imagine, there is a very quick multiplication of particles that happens in this showering process. The shower size continues to increase until the energy of the particles drops below some critical energy, at which point the

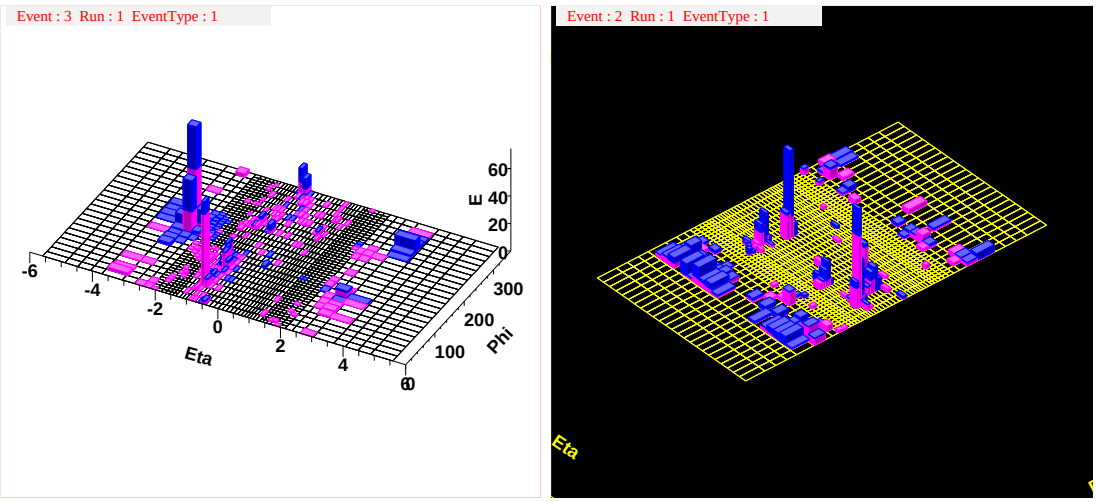


Figure 5.7: CDF Calorimeter Towers

primary mode of energy loss becomes ionization, and the shower decays. This is why the total number of particles in the shower is directly proportional to the incoming particle's energy.

In the hadronic calorimeters, the same basic idea is used to measure hadronic energies. In analogy to their EM counterparts, hadrons produce a shower of products, but now through the strong interaction with the absorbing material. The resulting particles in these showers are much more diverse. Because of the wide variety of interactions which can take place through the strong force, the resolution of the hadronic calorimeters is typically worse than the electromagnetic. Products such as muons and neutrinos leave the detector unseen, and events like neutron and proton capture make the picture much more confused.

Hadrons (*i.e.* quark final states: neutrons, protons, pions, *etc.*) have a much longer interaction length than their EM counterparts. The nuclear interaction length λ_0 (appropriate for hadronic showers) is much larger then the radiation length χ_0 used to characterize EM showers. To use the CDF systems as an example, the lead (Pb) of the EM calorimeters has $\chi_0 = 0.56$ cm, and the iron (Fe) of the hadronic sections has $\lambda_0 = 19$ cm. While hadrons do deposit energy within the EM calorimeters, their occupancy is clearly dominated by the EM particle contributions. Meanwhile, out at the radius of the hadronic systems, all of the EM particles will have showered away.

The central calorimeter sub-detector sections are divided into twenty-four (24) wedges that each cover 15° in ϕ. The individual wedges then have ten (10) towers

of 0.1 each in η. These towers have a projective geometry, by which it is meant that they point back toward the interaction point. The EM sections come first, and are closest to the beamline. Their scintillator and lead (Pb) sections are interrupted by an intermediate system called the CES (Central Electromagnetic Shower maximum) detector. Its purpose is to further localize and describe the geometry of the EM showers. At the shower maximum, the EM particles have dropped below critical energy, and the shower has peaked in size. The CES is a proportional chamber filled with 95% argon (Ar) and 5% carbon dioxide (CO_2), and it is located within the CEM at the shower maximum (approximately 6 χ_0). The position information this detector provides is used to help associate particle tracks to EM objects in the event reconstruction. The CES is then surrounded by the remaining CEM layers and the 32 hadronic layers of iron (Fe) and acrylic scintillator.

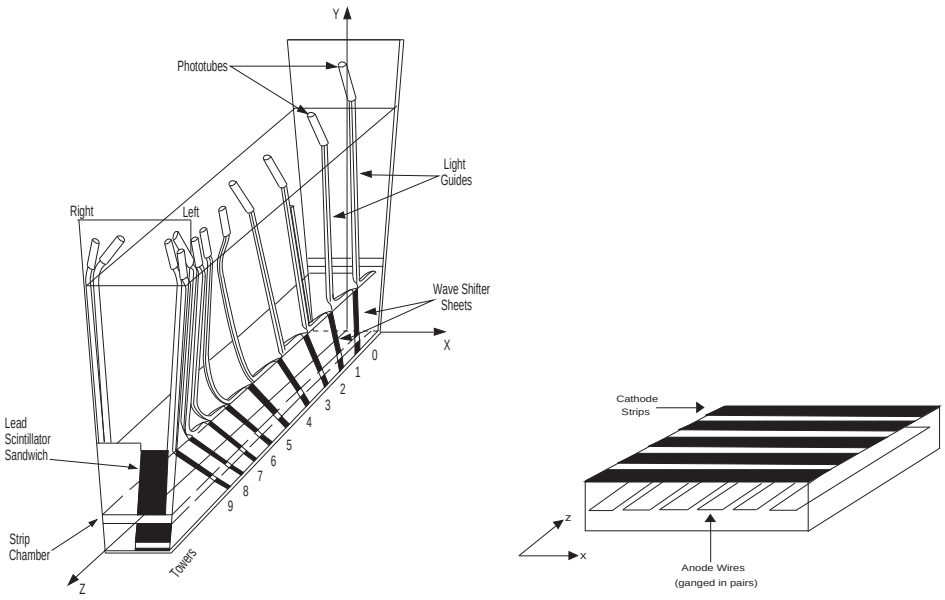


Figure 5.8: Central Calorimeter Wedge and CES

The plug systems are similar in conception to their central cousins. Just as the CES is embedded in the CEM, a detector called the PES (Plug Electromagnetic Shower) is located at $\sim 6\ \chi_0$ inside the PEM. Its purpose is identical to the CES, but it uses two layers of scintillator strips aligned at angles to provide the position information. See Figure 5.3 for a cross section of the plug calorimeters.

An interesting facet of these types of calorimeters is that their energy resolution actually increases (gets better) with the energy of the incoming particle. This is in

contrast to the tracking systems, in which the "stiffer" the track, the harder it becomes to accurately measure its curvature and place of origin.

Figure 5.7 shows a graphical representation of the calorimeter towers taken from the CDF control room's event display. In the picture, the longitudinal angle ϕ is unwrapped, and is the shorter edge of the overall rectangle. η is represented along the larger edge, and each square in the grid shows the EM and Hadronic energy occupancy for that tower in the event. The EM energy is shown in pink, while hadronic energy is blue.

See Figure 5.9 for a summary of the CDF calorimeters.

Detector	CEM	PEM	CHA	WHA	PHA
Coverage	$\|\eta\| < 1.1$	$1.1 < \|\eta\| < 3.6$	$\|\eta\| < 0.9$	$0.7 < \|\eta\| < 1.3$	$1.1 < \|\eta\| < 3.6$
Modules	48	24	48	48	24
η Towers/Module	10	12	8	6	10
Layers	31	23	32	15	23
Absorber	Lead	Lead	Iron	Iron	Iron
	3.2 mm	4.5 mm	2.5 cm	5 cm	5 cm
Scintillator	poly.	poly.	acrylic	acrylic	acrylic
	5 mm	4 mm	1 cm	1 cm	6 mm
Radiation Length	18 χ_0	21 χ_0	4.7 λ_0	4.5 λ_0	7.0 λ_0
Resolution(δE/E)	$14\%/\sqrt{E}$	$16\%/\sqrt{E}$	$75\%/\sqrt{E}$	$75\%/\sqrt{E}$	$80\%/\sqrt{E}$

Figure 5.9: CDF Calorimetry Summary

5.4 Muons & Neutrinos

With all the heavy calorimeters (purposefully) impeding the way toward freedom, the last detector systems at the outside of CDF are designed to register muons.

The only other particles expected to make it through all the detector material other than muons ($\mu^{\pm}$) are neutrinos (ν_e, ν_μ, ν_τ). Because neutrinos hardly interact with matter at all, there is really no hope of detecting them with a machine on the scale of CDF. Therefore, neutrinos are lumped into a category called "Missing Transverse Energy", or $\not\!\!E_T$. In the real world our detectors are imperfect. There are cracks between sub-detectors, which exist to provide for space for readout electronics and other essential systems such as cryogenics. Also, we do not have complete coverage over the entire solid angle surrounding the interaction point. We do our best to contend with all these things through efficient design methods. However, despite our best

efforts, neutrinos will always slip through our grasp and must be included with the unmeasurable.

Because protons and anti-protons are composite objects made of quarks and gluons, the overall z component of momentum for the initial collision objects need not net zero. This results from the fact that we can never know exactly what fraction of the proton's momentum in z is carried by the parts that interact in any specific collision event. Was that collision between a gluon and a down quark, and how much of the 1 TeV of energy did each possess? These questions aren't answerable with our detectors. On the other hand, the overall transverse components of momentum (p_x and p_y) before the $p\bar{p}$ collision do add to zero. Consequently, through conservation of momentum we expect that the vector sum of the transverse components of all the debris to exactly cancel as well. Therefore, it is a good first order assumption to conclude that any imbalance in E_T is due to neutrino emission.

The situation is not so hopeless for muons. Muons are quasi-stable. That is, with a lifetime of 2.2 μs, they live long enough to easily travel through the entire CDF detector. At the same time, their modest mass of 105.7 MeV/c^2 allows them to plow their way through the CDF calorimetry. In general terms, possessing a larger mass than the electron causes the muon to emit less bremsstrahlung radiation. This causes the showering for muons to be much reduced compared to electrons, and hence they deposit very little energy in the calorimeters. Compared with almost everything else produced in our collider, muons are able to penetrate large distances through normal matter. For this reason, it is the final layer of detection equipment surrounding CDF that is devoted to spotting muons.

The central systems for muon detection are called the Central Muon (CMU) and Central Muon Upgrade (CMP), sometimes collectively referred to as CMUP. Also present are the Central Muon Extension (CMX) and the Intermediate Muon System (IMU). Their coverages are shown in Figure 5.10. The CMUP covers $|\eta| \leq 0.6$, the CMX from $0.6 < |\eta| \leq 1.0$, and the IMU covers $|\eta| > 1$. The CMUP covers the full 360° in ϕ. In both the CMX and IMU, there are gaps in ϕ which are due to engineering and the physical constraints of other detector components. As of this writing, the IMU and its associated forward muon detection systems have not been fully integrated into the standard CDF event reconstruction process. Muons from this η range are usually referred to as barrel muons (BMU). While no forward muons were used in the analysis presented in this thesis, some sections may make reference to them.

Much like the COT, the muon systems are based on the drift chamber principle.

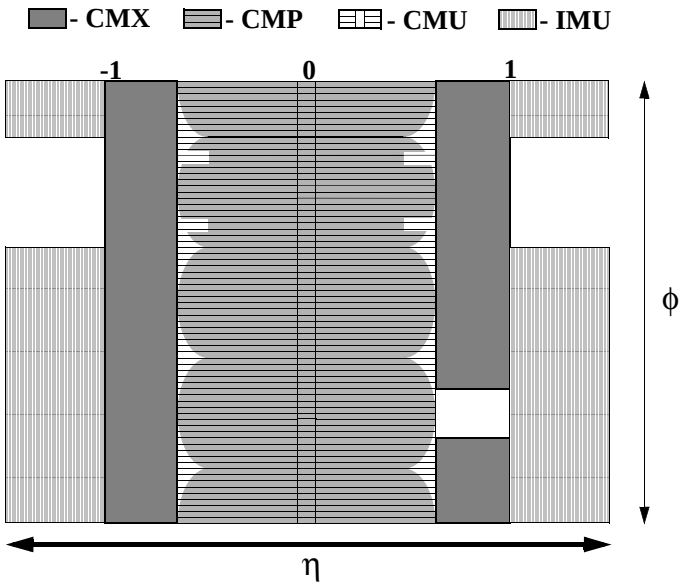


Figure 5.10: Muon System η - ϕ Coverage

Chambers consisting of single sense wires are layered, with adjacent layers offset in ϕ to provide higher resolution. The CMU has a total of four (4) layers. When a hit is registered within three or more (≥ 3) layers, we have a so-called muon "stub" at that location. This essentially means there is enough information to reconstruct a small muon track (hence stub) at that spot. Located behind the CMU, the CMP contains another set (4 offset layers) of single wire drift chambers. While only furnishing ϕ information, the CMP layers provide for increased discriminatory power when added to existing stubs within the CMU. Finally, outside the CMU and CMP is a layering of scintillation counters. These yield timing data, which aids in muon track extrapolation from the Silicon and COT systems.

Once outside the "block" of the central calorimeters ($|\eta| > 0.6$), muon coverage is taken over by the CMX. As is shown in Figure 5.12, the CMX is built up to form arcs of layered drift chambers called "arches". The photograph in Figure 5.12 shows the arches (the semi-circular silver strips) during their installation. Figure 5.11 shows their placement in a slice through the x-z plane, and a detailed view of the layering scheme used. The arches are installed on the "sides" of CDF, while the top and bottom gaps between the arches are filled by the keystone and miniskirt. Note that the miniskirt is only operational on the East side of CDF, and accounts for the gap in ϕ shown in Figure 5.10. Just as with the CMUP, the CMX is completed with a scintillation layer

(CSX).

One may wonder about tau leptons ($\tau^{\pm}$), and how they manifest themselves in a machine like CDF. Though possessing a relatively high mass (1.777 GeV/c^2), taus have a very short lifetime of only 2.9×10^{-13} s. This allows them to travel some 100 μm within the CDF detector before decaying. This can happen hadronically ($\sim$65%) or leptonically. Here are some examples:

$$\begin{aligned} \tau^{\pm} &\rightarrow e^{\pm}\nu_e\nu_\tau \\ \tau^{\pm} &\rightarrow \mu^{\pm}\nu_\mu\nu_\tau \\ \tau^{\pm} &\rightarrow \pi^{\pm}\nu_\tau \\ \tau^{\pm} &\rightarrow \rho(\pi^{\pm}\pi^0)\nu_\tau \rightarrow \pi^{\pm}\gamma\gamma\nu_\tau \end{aligned} \tag{5.2}$$

The point is that tau leptons quickly become objects which our detector can handle via the systems described previously. Taus are discernible via their large invariant mass, large impact parameter, and the missing transverse energy from their decay product neutrinos. Final states of hadrons and photons from π's are similarly characteristic. See [23] for a summary of tau detection and identification at the Tevatron.

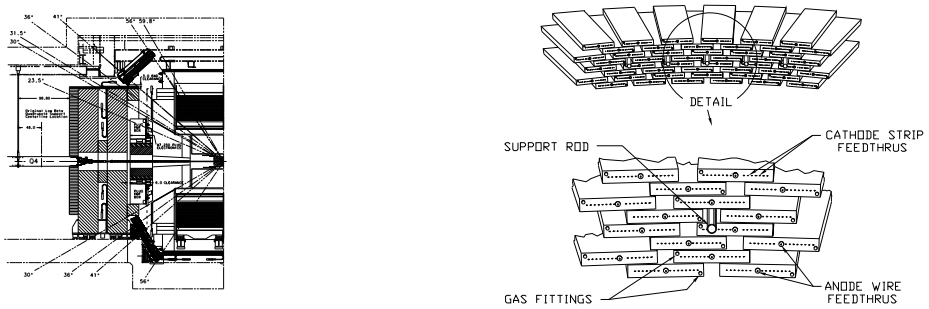


Figure 5.11: Central Muon Extension (CMX)

5.5 Luminosity

Another important quantity for any hadron collider experiment to measure is how many collision events have actually been delivered to the detectors. This information can be addressed through what is known as the time integrated luminosity for the experiment. Luminosity in many contexts usually connotes brightness, and here one

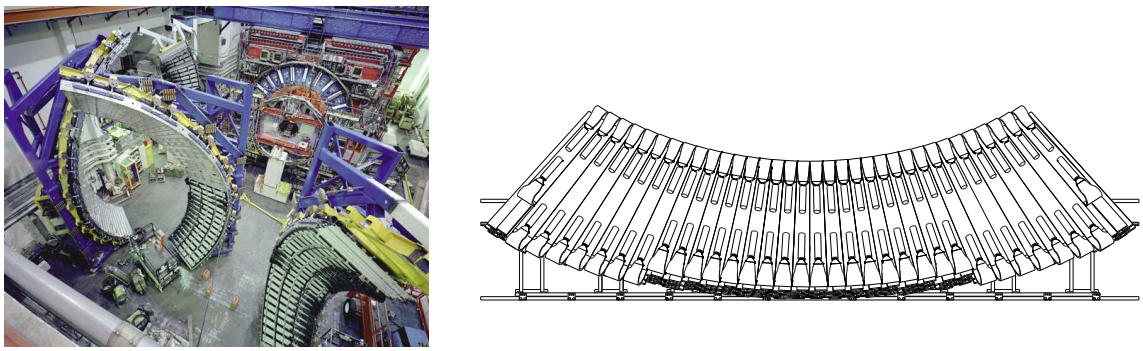


Figure 5.12: CMX Arches & Miniskirt

Detector	CMU	CMP	CMX
Coverage	$\|\eta\| < 0.6$	$\|\eta\| < 0.6$	$0.6 < \|\eta\| < 1.0$
Drift tube size	2.68×6.35×226 cm	2.5×15×640 cm	2.5×15×160 cm
Max drift time	0.8 μs	1.4 μs	1.4 μs
Scin. thickness		2.5 cm	1.5 cm
Total Counters		269	324
Min muon p_t	1.4 GeV/c	2.2 GeV/c	1.4 GeV/c

Figure 5.13: CDF Muon Summary

might be led to ask the "brightness" of the interaction point of the colliding beams. In particle and collider physics, this word is used to describe the potential number of interactions per unit time. Specifically, luminosity is the proportionality factor between production cross section σ and event rate. For a given process x, the number of events of that process generated per unit time is given as:

$$\frac{N_x}{dt} = \sigma_x \mathscr{L}_{inst} \tag{5.3}$$

Instantaneous luminosity is recorded in units of (area time)$^{-1}$, and is some large number. A typical peak luminosity value for a given run at the Tevatron is 5×10^{31} cm^{-2} s^{-1}. When speaking of the time integrated value of luminosity, particle physicists often find it useful to introduce the following units.

$$1 \text{ barn (b)} = 10^{-24}\text{cm}^2 \tag{5.4}$$

Walking around Fermilab, one might overhear some physicists talking about the latest 300 inverse picobarns of data. We often use time integrated luminosity to describe the size of a dataset, or the amount of colliding particles which have been delivered to

the detectors. Since cross sections are usually quoted in barns, it is convenient to speak of luminosity in terms of inverse barns. The origin of the barn unit dates back to World War II. The February 2006 issue of *Symmetry*[9] magazine describes its inception.

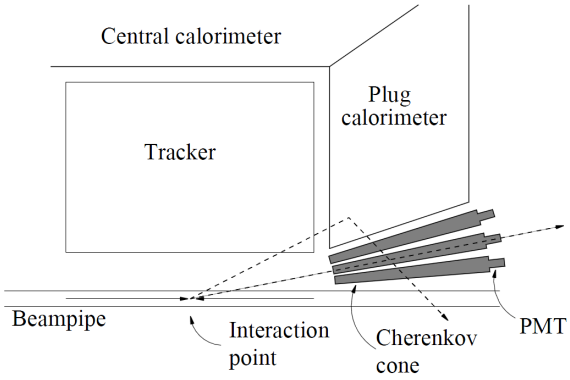


Figure 5.14: CLC Schematic View

Luminosity is measured at CDF through the use of Cherenkov Luminosity Counters (CLC). These counters sit just off the beamline in a conical configuration located at polar angle $\theta = 3^\circ$ (see Figure 5.14). For Run II, these counters measure luminosity by sampling the beam crossings at 1 Hz. For each sample, the counters decide whether (or not) there has been inelastic scattering. The following equation is then used to calculate the time integrated luminosity:

$$\int_{\Delta t} (\mathscr{L}_{inst}) dt = \frac{N_X}{\sigma_{p\bar{p}} \cdot N_B} \sum_{j=1}^{N_B} \frac{N_{p\bar{p}}^j}{N_S^j} \tag{5.5}$$

Here, N_X is the number of bunch crossings during time Δt, N_B is the number of bunches in the Tevatron, $\sigma_{p\bar{p}}$ is the proton / anti-proton inelastic cross section, N_S the number of trigger samples taken by the CLC, and $N_{p\bar{p}}$ the number of inelastic collisions observed by the CLC. See CDF Notes 4831 [6] and 6052 [7] for a more detailed discussion.

5.6 Trigger

All of this sophisticated hardware designed to detect the high energy debris from our $p\bar{p}$ collisions would be useless if we were unable to read out the information they provide in a timely fashion. As has been previously stated, the Tevatron has a bunch crossing

every 396 nanoseconds in its Run II operation mode. This gives a very short window of opportunity to look at the collisions that have taken place, determine if they are somewhat interesting, and if so record all the information our detector systems make available.

The trigger system is responsible for this task at CDF. It consists of hardware designed to make decisions about which collisions to record. A 396 nanosecond bunch crossing translates into approximately 2.5 million crossings per second. Our data acquisition system (DAQ) is only capable of recording data from the detector elements at a rate on the order of 50 Hertz.

This is not as daunting a situation as it may seem at first glance. The vast majority of collision events in a collider like the Tevatron are very similar. What are known as "minimum bias" events account for a large fraction of possible outcomes. These are events where no hard scattering takes place, and the proton and anti-proton can be said to ooze through one another. Most of the debris continues along the beamline, and the detector occupancy is minimized. By way of comparison, the total $t\bar{t}$ cross section is nine orders of magnitude smaller than the minimum bias cross section[16].

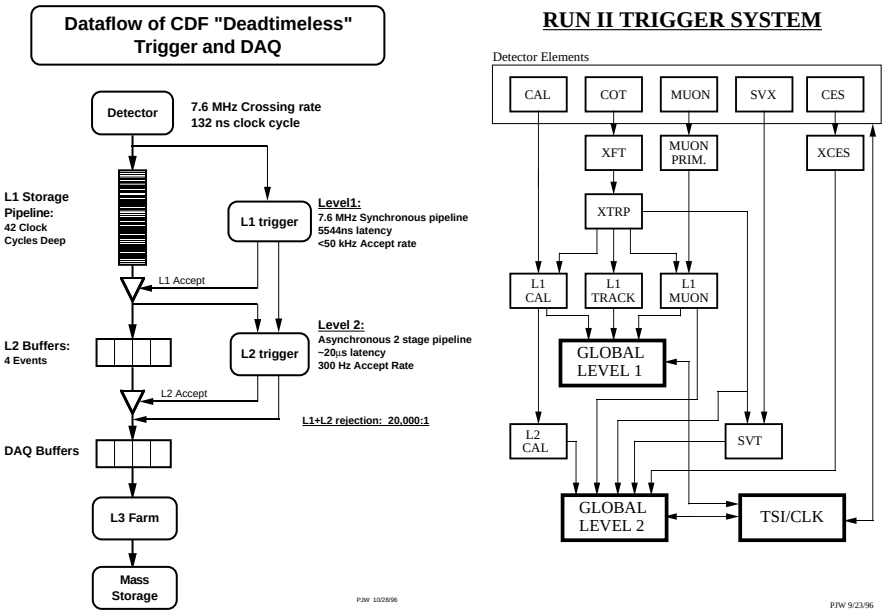


Figure 5.15: CDF Run II Data Flow and Trigger Schematic

5.6.1 Trigger Hierarchy

The trigger system at CDF uses a three tiered protocol, where each pass is designed to separate out the interesting events from those more mundane.

The first level of decision hardware consists of custom electronics designed to use only a subset of the detector information. It receives information from the calorimeter, central tracking chamber, and muon detectors. The decisions it makes are based on counting simple objects (*i.e.* muon, electron, or jet candidates) from these sub-systems. For example, we might want to pass events that contain a 10 GeV electron, or perhaps others with two 1.5 GeV muons. See Figure 5.15 for a block diagram of the overall trigger system's data flow.

Level 1 is pipelined for 42 beam crossings. This means that it can store the information from 42 crossings while waiting for the higher level triggers to accept and process the data which has been forwarded to them. These data pipelines exist at all trigger levels, and are designed to minimize "dead-time". That is to say, we would like to avoid losing collision events because our trigger hardware was overloaded and couldn't process them.

Level 2 further reduces the acceptance rate by a factor of 160, moving from something near 50 kHz at Level 1 down to 300 Hertz. Level 2 is an asynchronous system which has four event buffers, and takes approximately 20 μs to make a decision. Upon a L2 acceptance, the event data is passed along to the DAQ buffers for reconstruction at Level 3. L2 allows for more complex event primitives to be used as discriminators. All L1 data is available at L2 (though some with more precision), and having had extra time to calculate, some additional information is also in evidence. The most notable of these additions to the L1 features is the ability to use output from the SVT at L2. The silicon detector allows for the identification of tracks with displaced vertices (*e.g.* which indicate b quark production). Also, the L2 cluster finder allows for more accurate jet based triggers. Overall, decisions at this level can be based on the presence of displaced tracks, the event's missing transverse energy, and the number, and/or energy of, any electrons, muons, photons, and jets.

Level 3 hardware is made up of a dedicated CPU farm located on the top floor of the CDF complex. It is used to fully reconstruct the event fragments passed along from L2. The L3 event reconstruction takes advantage of the full detector information, including some pieces not available to the other trigger levels. This allows for full 3-dimensional track reconstruction, and complete matching of tracks to the calorimeters. Once an

event has been duly considered and found desirable, its characteristics are passed off to the mass storage system. Level 3 has a target acceptance rate of 50 Hertz, with the average size of an event being approximately 150 kilobytes. This translates to roughly 7.5 MB/s of recordable data during acquisition, with peaks in the neighborhood of 15 MB/s.

5.6.2 Trigger Paths

Events that end up in mass storage for offline analysis consequently have followed some logical path trough the trigger system. The triggers used in the analysis presented in this thesis came from stream B, which triggers on electrons and muons that have large transverse momenta (p_T) in the central region of the detector. The details of these triggers have evolved over the course of Run II. The revisions made were motivated by the rise in the initial instantaneous luminosity the Tevatron experienced as the machine's performance was improved, changes in detector hardware performance over time, and our improved ability to trigger efficiently.

Level	CEM	CMUP	CMX
L1	L1_CEM8_PT8_v-4	L1_CMUP6_PT4	L1_CMX6_PT8_PS1 L1_CMX6_PT8_CSX_PS1
L2	L2_CEM8_PT8_CES3_TRK5_DPHI10 L2_CEM8_PT8_CES8_TRK8	L2_AUTO_L1_CMUP6_PT4 L2_TRK8_L1_CMUP6_PT4	L2_AUTO_L1_CMX6_PT8 L2_AUTO_L1_CMX6_PT8_CSX
L3	L3_ELECTRON_CENTRAL_18	L3_MUON_CMUP18	L3_MUON_CMX18

Figure 5.16: Example Stream B Trigger Paths

To provide a flavor for the trigger requirements involved in stream B, Figure 5.16 lists some examples. The figure shows trigger paths for CEM electrons and CMUP/CMX muons. Numbers following the sub-system's text (*i.e.* CMUP6) are the minimum energy requirement for that object. The number following PT is the transverse momentum needed in the lepton primitive, and TRK lists the transverse momentum cut on the associated track. AUTO indicates the trigger is automatically accepted at that level, and PSN indicates a "pre-scaled" trigger (*i.e.* PS3 would mean only every 3^{rd} event which passes is actually forwarded to the next level). Individual triggers at each level are combined with a logical OR operation, while requirements across levels are AND'ed together.

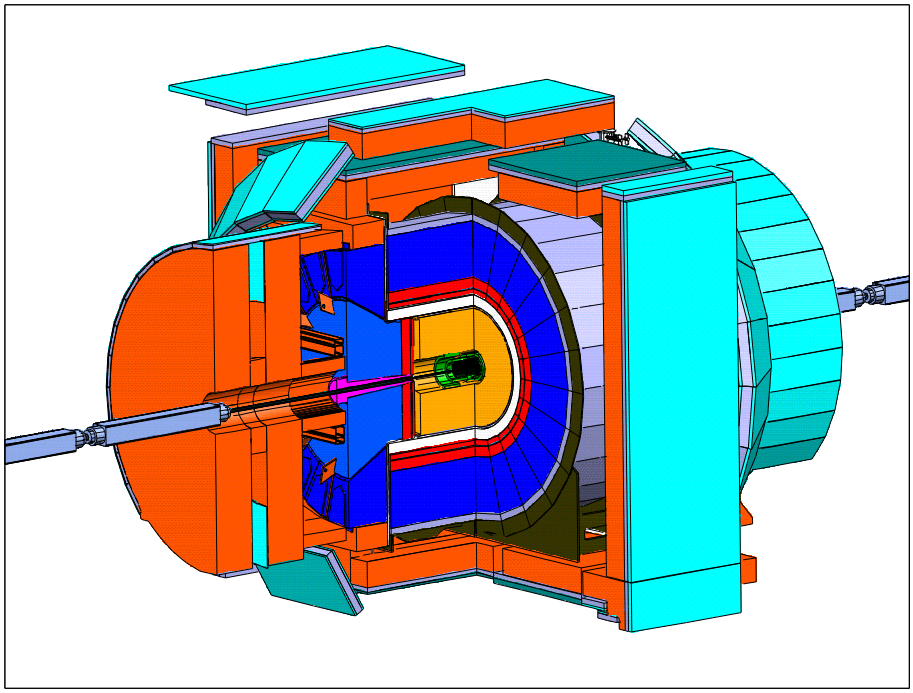


Figure 5.17: CDF

Part IV

Analysis

Chapter 6

Problem Description

As a physics analysis, the SVM formalism described in the opening chapters has been used to measure the $t\bar{t}$ production cross section ($\sigma_{t\bar{t}}$) at the Tevatron. This methodology has been applied to the so-called lepton plus jets decay channel (L+J), where one half of the $t\bar{t}$ system decays leptonically, while the other goes out hadronically. The signature of this type of decay in the CDF detector is 1 high transverse momentum ($\vec{p}_T$) lepton, and four (4) hadronic jets. The primary Feynman diagram for our signal process is given in Figure 6.1. Processes that mimic this signature are mainly W's with QCD jets. Figure 6.2 depicts a representative example of W production at the Tevatron.

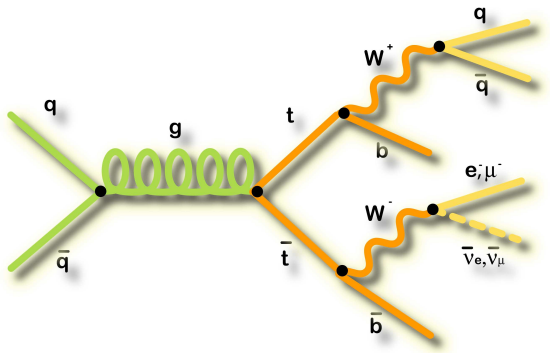


Figure 6.1: Feynman diagram for decay of $t\bar{t}$ to lepton plus jets

The measurement of $\sigma_{t\bar{t}}$ can be broken down into several steps. The major brush strokes of the analysis process are given here, so an overarching vision can be kept in mind by the reader.

First, we will derive an equation which will be used to calculate $\sigma_{t\bar{t}}$. This expression

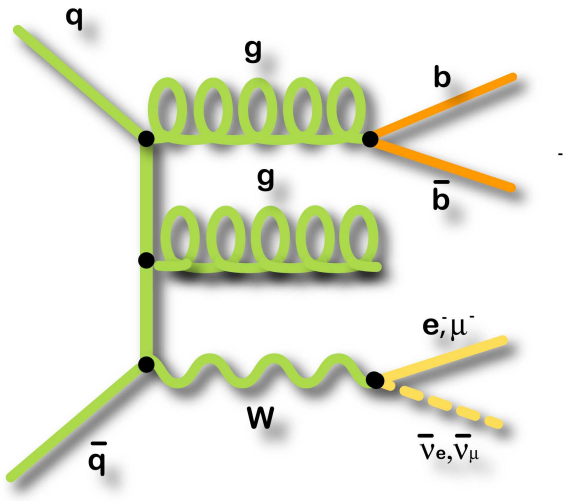


Figure 6.2: Example Feynman diagram for W + jets production

will include scale factors for dealing with corrections to individual terms, in order that differences between simulation and the actual performance of the CDF detector may be included. Also, the CDF detector will be treated logically as three (3) separate machines in this equation. Because we are using the L+J decay channel, it is possible to partition events by the type of lepton they contain. The three reconstructed lepton types which are kept in this analysis are CEM electrons, CMUP muons, and CMX muons. The trigger and scale factors for the CMUP and CMX are sufficiently different such that a separate treatment in our derivation is warranted.

Next, we will lay out the specifics of our event selection criteria, and we will talk briefly about the high level objects used in CDF's event reconstruction. The results of using this event selection to calculate acceptances and scale factors will be listed.

The details of how we compose event vectors for the SVMs will then be discussed. This includes any "post processing" done on events. That section will conclude with the training procedure used for the SVM systems, and the resulting performance characteristics obtained.

The trained SVM framework will be used to construct pseudo-experiments from Monte Carlo events, which are meant to mimic the actual measurement of $\sigma_{t\bar{t}}$ when using CDF's real world data. These pseudo-experiments will allow us to quantify many sources of uncertainty on our final value for the cross section. These sources of error will then be detailed.

Finally, the SVM framework will be used to calculate $\sigma_{t\bar{t}}$ in the same way as was done in the pseudo-experiments, and our measured value will be quoted with appropriate errors.

6.1 Cross Section

In basic particle physics theory, the idea of production cross section is usually defined to represent the likelihood of creating a specific physics object during some type of collision event.

$$\sigma_x = \frac{N_x}{\mathscr{L}} \tag{6.1}$$

where N_x is the number of type x events, $\mathscr{L}$ is the time integrated luminosity, and σ_x is the production cross section for that process.

Suppose we had a large sample of collision events which corresponded to an integrated luminosity $\mathscr{L}$, and we desired to know the $t\bar{t}$ production cross section $\sigma_{t\bar{t}}$. We further suppose that these collision events were observed with an imperfect detector, and using this detector we were able to record some of these events which we thought would be interesting to study. After some thought, we devise a way to distinguish $t\bar{t}$ events from other types within the observed N_{obs} events we saved. This technique calculates the fractional number of events it believes to be $t\bar{t}$ within a given sample. We will call this fractional value θ. Because our detector is imperfect and only covers a finite solid angle about the collision point, we will miss and be unable to record some $t\bar{t}$ events. There are electronic effects associated with the detector's readout and trigger which also cause us to lose some events of interest. Taken together, we can express this loss as an *efficiency* of our apparatus. Let us define $\varepsilon_{t\bar{t}}$ as the amount of $t\bar{t}$ that is kept while observing collisions with our detector, as a fraction of what potentially could have been seen. If, after the detector effects and trigger loss has taken place, we were to lose 35% of the $t\bar{t}$ events, then $\varepsilon_{t\bar{t}} = 0.65$. If there are some quality cuts or an event selection criteria applied in the compilation of the N_{osv} data sample, then these effects should also be accounted for in this ε factor. Our expression for the $t\bar{t}$ production cross section becomes

$$\sigma_{t\bar{t}} = \frac{N_{t\bar{t}}}{\mathscr{L}} = \frac{(\theta N_{obs})\varepsilon_{t\bar{t}}^{-1}}{\mathscr{L}} \tag{6.2}$$

as (θN_{obs}) gives the number of $t\bar{t}$ events in our observed data set, and dividing by

the efficiency takes us back to the total number of $t\bar{t}$ events we could have seen with a perfect detector.

In practice, the detectors we use are broken down into several specialized subsystems. Each of these systems records events based on a trigger, and thus have corresponding trigger efficiencies for each of the different types of events one might like to examine. So, a typical calculation of the $\sigma_{t\bar{t}}$ across detector elements might look as follows

$$N^{obs} = \varepsilon_{detector} N^{all} \tag{6.3}$$

N^{obs} is the number of observed events from some process in a detector element, while ε is that detector's efficiency for that process, and N^{all} is the total number of events from that process which would have been seen in a perfect detector.

$$N^{obs} = N^{obs}_{CEM} + N^{obs}_{CMUP} + N^{obs}_{CMX} \tag{6.4}$$

$$N^{obs} = \varepsilon_{CEM} N^{all}_{CEM} + \varepsilon_{CMUP} N^{all}_{CMUP} + \varepsilon_{CMX} N^{all}_{CMX} \tag{6.5}$$

Assuming the process of interest is $t\bar{t}$:

$$N^{obs}_{t\bar{t}} = \varepsilon_{CEM}(\sigma_{t\bar{t}} \mathscr{L}_{CEM}) + \varepsilon_{CMUP}(\sigma_{t\bar{t}} \mathscr{L}_{CMUP}) + \varepsilon_{CMX}(\sigma_{t\bar{t}} \mathscr{L}_{CMX}) \tag{6.6}$$

which implies:

$$\sigma_{t\bar{t}} = \frac{N^{obs}_{t\bar{t}}}{\varepsilon_{CEM}\mathscr{L}_{CEM} + \varepsilon_{CMUP}\mathscr{L}_{CMUP} + \varepsilon_{CMX}\mathscr{L}_{CMX}} \tag{6.7}$$

Each of the ε terms in (6.7) would be obtained from Monte Carlo studies of the process in question, and a simulation of the detector subsections. Trigger efficiencies, obtained in this same manner, modify the individual terms. Making the assumption that there are certain detector effects that are common to all detector elements, an ε_{common} can be factored out. Taking $N^{obs}_{t\bar{t}} = \theta N_{obs}$, our final expression looks as follows:

$$\boxed{\sigma_{t\bar{t}} = \frac{\theta N_{obs}}{\varepsilon_{common}(\varepsilon_{CEM}\mathscr{L}_{CEM} + \varepsilon_{CMUP}\mathscr{L}_{CMUP} + \varepsilon_{CMX}\mathscr{L}_{CMX})}} \tag{6.8}$$

where we will take (*e.g.*):

$$\varepsilon_{CEM} = SF^{CEM}_{id} \times SF^{CEM}_{trig} \times \mathcal{A}_{CEM} \tag{6.9}$$

with SF_{id} and SF_{trig} being identification and trigger scale factors, $\mathcal{A}$ the raw Monte Carlo acceptance, and ε_{common} to include any factors applicable to all detector elements.

The following sections will detail these efficiency factors for our final calculation of the $t\bar{t}$ cross section.

6.2 Sub-detector Efficiencies

The sub-detectors we have used in this analysis to identify event candidates are the Central Electromagnetic (CEM), Central Muon (CMUP), and Central Muon Extension (CMX). We use these detector elements to identify high transverse momentum (p_T) leptons, employing the CEM to find electrons, and the CMUP/CMX for muons. These are the main central detectors, and cover $|\eta| \leq 1.5$ for CEM, $|\eta| \leq 0.6$ for CMUP, and $0.6 < |\eta| \leq 1.2$ for CMX. The PHX (plug a.k.a. phoenix electrons) and BMU (barrel muon) detectors, which cover larger values in $|\eta|$, are not used in this analysis.

Detector	Potential	Saved	Acceptance	Stat Error
CEM	4,461,497	199,231	0.044656	±0.000098
CMUP	4,461,497	125,425	0.028113	±0.000078
CMX	4,412,909	53,785	0.012188	±0.000052
PHX	4,461,497	50,482	0.011315	±0.000050
BMU	4,461,497	32,115	0.007198	±0.000040

Figure 6.3: Raw Monte Carlo Acceptance

The starting basis for the efficiency factors for the CEM, CMUP, and CMX for the $t\bar{t}$ process come from the raw Monte Carlo acceptance values for each of these sub-detectors. CDF has a large pool (∼5 Million) of simulated $t\bar{t}$ decays, generated with a top quark mass of 175 GeV/c^2. This pool is all inclusive in its decay modes, meaning that every possible decay of the $t\bar{t}$ system is represented as predicted by the Standard Model. These Monte Carlo simulated events have been passed through a realistic detector simulation. This simulation produces output that mirrors that of the real data coming from the actual CDF detector. After running over these events with our event selection, and recording the number of CEM, CMUP, and CMX events we retain, a first order estimate of the acceptance is obtained. These values have been recorded in Figure 6.3. PHX and BMU are also listed there for reference.

Each of these sub-detectors has an associated "scale factor" that corrects the Monte Carlo simulation to fit the real data with regard to the identification and reconstruction

Period	StartRun	EndRun	Lumi[pb^{-1}]	CMX Lumi[pb^{-1}]
0d	138425	186598	331.47	318.11
0h	190697	203799	362.94	359.50
0i	203819	212133	258.37	258.37
8	217990	222426	166.29	166.29
9	222529	228596	156.76	152.78
10	228664	233111	243.56	243.49
11	233133	237795	234.99	229.98
12	237845	246231	163.96	157.13
13	241665	246231	280.63	268.17
All	138425	246231	2,198.97	2,153.82

Figure 6.4: Luminosity for Silicon Good Run List v20

of the electron and muon objects. It has been found that usually the simulation is more perfect than the real detector in picking out high P_T leptons. These correction terms are called the "idreco scale factors" in CDF notes and documentation, and represent the data idreco rate divided by the simulated rate. These scale factors are calculated over each run period of the experiment at the Tevatron to account for any changes that may occur in the circumstances of the detector. In order to combine these terms over the complete run history, their values have been averaged, weighting each scale factor by the delivered luminosity during its corresponding run period. The time integrated luminosity for the different run periods used in this analysis are shown in Figure 6.4. The terminology "silicon good run list" indicates that the list of validated datasets used to compute the total delivered luminosity includes only those runs that had CDF's silicon tracking system activated. The following three (3) tables detail the idreco scale factors (SF_{id}) for CEM, CMUP, and CMX. The final row labeled "All" shows the luminosity weighted value to be used in Equation 6.8 when calculating $\sigma_{t\bar{t}}$.

Period	StartRun	EndRun	CEM SF	Error
0d	138425	186598	0.991	±0.004
0h	190697	203799	0.985	±0.004
0i	203819	212133	0.974	±0.004
8	217990	222426	0.977	±0.006
9	222529	228596	0.978	±0.006
10	228664	233111	0.978	±0.005
11	233133	237795	0.968	±0.007
12	237845	246231	0.961	±0.007
13	241665	246231	0.975	±0.005
All	138425	246231	0.978	±0.005

Figure 6.5: CEM Electron Identification & Reconstruction Scale Factors

Period	StartRun	EndRun	CMUP SF	Error
0d	138425	186598	0.936	±0.0055
0h	190697	203799	0.929	±0.005
0i	203819	212133	0.917	±0.0064
8	217990	222426	0.931	±0.0078
9	222529	228596	0.940	±0.008
10	228664	233111	0.932	±0.007
11	233133	237795	0.899	±0.008
12	237845	246231	0.919	±0.009
13	241665	246231	0.924	±0.007
All	138425	246231	0.9253	±0.0068

Figure 6.6: CMUP Muon Identification & Reconstruction Scale Factors

Period	StartRun	EndRun	CMX SF	Error
0d	138425	186598	1.0098	±0.0057
0h	190697	203799	0.9744	±0.007
0i	203819	212133	0.9781	±0.009
8	217990	222426	0.9704	±0.012
9	222529	228596	0.9565	±0.013
10	228664	233111	0.9655	±0.011
11	233133	237795	0.9456	±0.011
12	237845	246231	0.9577	±0.014
13	241665	246231	0.9466	±0.012
All	138425	246231	0.9697	±0.0098

Figure 6.7: CMX Lepton Identification & Reconstruction Scale Factors

The following three (3) tables list the trigger efficiencies (SF_{trig}) for CEM, CMUP, and CMX.

Period	StartRun	EndRun	CEM Eff	Error
0d	138425	186598	0.962	±0.007
0h	190697	203799	0.976	±0.006
0i	203819	212133	0.979	±0.004
8	217990	222426	0.959	±0.007
9	222529	228596	0.960	±0.002
10	228664	233111	0.959	±0.002
11	233133	237795	0.961	±0.004
12	237845	246231	0.960	±0.003
13	241665	246231	0.957	±0.003
All	138425	246231	0.9647	±0.0044

Figure 6.8: CEM Electron Trigger Efficiency

Period	StartRun	EndRun	CMUP Eff	Error
0d	138425	186598	0.9015	±0.0044
0h	190697	203799	0.9191	±0.0039
0i	203819	212133	0.9179	±0.0048
8	217990	222426	0.913	±0.0061
9	222529	228596	0.927	±0.0067
10	228664	233111	0.8655	±0.0072
11	233133	237795	0.8623	±0.0098
12	237845	246231	0.8417	±0.012
13	241665	246231	0.8284	±0.0091
All	138425	246231	0.8871	±0.0067

Figure 6.9: CMUP Muon Trigger Efficiency

Period	StartRun	EndRun	CMX Eff	Error
0d	138425	186598	0.9665	±0.0036
0h	190697	203799	0.9120	±0.0063
0i	203819	212133	0.9049	±0.0079
8	217990	222426	0.8839	±0.0087
9	222529	228596	0.8659	±0.0084
10	228664	233111	0.9294	±0.0089
11	233133	237795	0.9272	±0.0098
12	237845	246231	0.9002	±0.0140
13	241665	246231	0.8926	±0.0094
All	138425	246231	0.9141	±0.0080

Figure 6.10: CMX Muon Trigger Efficiency

Figure 6.11 lists the scale factor resulting from the exclusion of events with a primary z-vertex located more than ± 60 cm in z from the detector's geometric center. This factor is common across all sub-detectors.

Period	StartRun	EndRun	z-vertex SF	Error
0d	138425	186598	0.951	± 0.005
0h	190697	203799	0.960	± 0.005
0i	203819	212133	0.966	± 0.005
8	217990	222426	0.968	± 0.002
9	222529	228596	0.968	± 0.002
10	228664	233111	0.968	± 0.002
11	233133	237795	0.968	± 0.002
12	237845	246231	0.972	± 0.001
13	241665	246231	0.972	± 0.001
All	138425	246231	0.9647	± 0.003

Figure 6.11: z-vertex Scale Factor (ε_{common})

Chapter 7

Event Selection

In order to start any type of high energy physics analysis in a detector such as CDF, one of the first things done is to abstract certain types of physics objects from the raw detector readout. Rather then play with things like calorimeter towers, tracks, and muon stubs, we would prefer the familiar concepts of electrons, muons, photons, and jets.

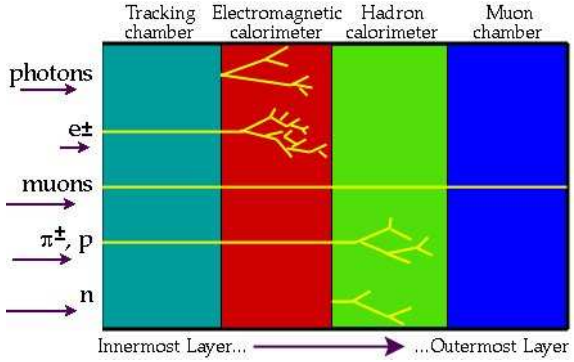


Figure 7.1: Decay Signatures

Figure 7.1 gives the basic signatures used to identify these objects at CDF Run II. Charged particles will leave tracks in the most central detector elements. For CDF this means the silicon detectors and the COT will provide us with information about the momentum of electrons, muons, and charged hadrons. The silicon detectors give a highly accurate impact parameter, or initial point of origin for a track, which can then be aligned with tracks from the COT. The curvature of the track through the magnetic field of the solenoid gives us a measurement of the particle's 3-momentum. It is important to measure this momentum before the particles start interacting with the

more invasive components of the detector. The calorimeters, while providing us with the necessary information about a particle's energy, are inherently active instruments that do not preserve the initial state of the incoming matter.

Electrons exhibit tracks in conjunction with EM energy which doesn't penetrate the EM calorimeter. Muons plow through everything, leaving small quantities of energy in the calorimeters as minimum ionizing particles. Uncharged particles such as neutrons or photons don't leave tracks in the COT, as they are unable to induce any currents in the sense wires. However, they can be told apart by looking at the type of energy they deposit in the calorimeters.

The specific requirement for a CEM electron are listed in Figure 7.3, while CMUP and CMX muons are described in Figure 7.4.

Jets are somewhat curious objects, as there is a greater degree of subjectivity in their definition. They conceptually are used to represent collections of hadronic and EM energy located in close (η, ϕ) proximity. Typically, they arise due to the hadronization of quarks and gluons into mesons and baryons. Jets are constructed by some algorithm which normally takes at least 1 parameter (the cone size $=\sqrt{\Delta\eta^2 + \Delta\phi^2}$). The standard algorithm used for CDF's event reconstruction is called JetClu-0.4. See [32] for a complete description of the JetClu method.

Since we are looking in the $t\bar{t}$ lepton plus jets (L+J) decay channel, we want an overall event topology similar to the following. According to the Standard Model, a top quark decays into a W boson and b quark about 99% of the time ($t \to bW^+$, $\bar{t} \to \bar{b}W^-$). Bottom quarks quickly decay (on the order of 10^{-12} seconds, or an approximate length of a few millimeters in the detector), and create hadronic jets. The W bosons can decay either into a lepton neutrino pair ($W^\pm \to \ell^\pm\nu$), or as hadrons ($W \to qq$). The case where both W's decay into leptons is called the dilepton (DIL) channel for $t\bar{t}$. The case where both W's go into hadrons is (suitably) called the all-hadronic channel. The case we have interested ourselves in for this analysis is the case where one W goes into a lepton/neutrino (L+J), and the other into hadrons (L+J). See Figure 6.1.

Thus, the basic event selection employed in this analysis is to ask for exactly 1 tight electron or muon, three or more tight jets, and a missing $E_T > 20$ GeV representing the lost neutrino. We loosen our selection to require only three jets (and not the 4 jets that might be expected reading directly off the Feynman diagram) to allow a slightly greater acceptance, and to make allowance for the loss of a jet down the beamline, event mis-reconstruction, or jet cone overlap.

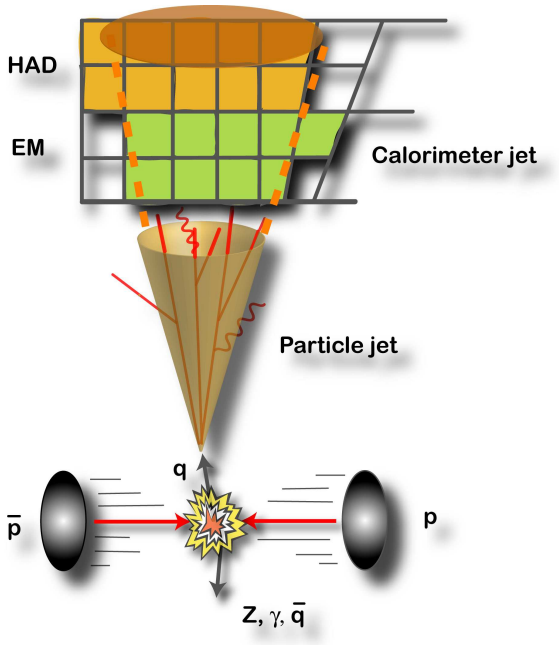


Figure 7.2: Schematic Depiction of Jets

7.1 High Level Objects

7.1.1 Electrons

There are specific "cuts" applied to the general tracking and calorimetry quantities from the detector that are used in order to identify electrons. This analysis employs CDF's most conservative cuts on the detector primitives to enumerate "tight" CEM electrons. Figure 7.3 lists the selection criteria.

- E_T: The transverse EM energy. Electrons are expected to expend the vast majority of their energy in the EM calorimeter. The E_T used here is the total EM energy in a "cluster" of calorimeter towers, times $\sin\theta$, where θ comes from the best COT track pointing to that cluster. Clusters for electrons are defined to be at most 3 towers: a seed tower pointed to by the track, and possibly 1 adjacent tower in ϕ and another in η.

- P_T: The transverse momentum of the best COT track from its curvature in the magnetic field.

- Iso4: This quantity called "isolation" is the amount of energy in a cone of radius

Quantity	Cut
E_T	> 20 GeV
p_T	> 10 GeV/c
Iso4	< 0.1
E_{HAD}/E_{EM}	$< 0.055 + 0.00045\ (E_{HAD} + E_{EM})$
E_T/p_T	< 2.0 or $p_T > 50$ GeV/c
L_{SHR}	< 0.2
$Q * \Delta x$	-3.0 cm to $+1.5$ cm
$\lvert\Delta z\rvert$	< 3.0 cm
χ^2_{strip}	< 10
$\lvert z_0\rvert$	< 60.0 cm
COT quality	3 axial SL & 2 stereo SL w/ ≥ 7 hits each SL

Figure 7.3: Tight CEM electron cuts

$R \equiv \sqrt{\Delta\eta^2 + \Delta\phi^2} \leq 0.4$ around the EM cluster, excluding the EM energy in that cluster, divided by that EM energy. It is therefore a measure of the energy in the neighborhood of the cluster, as a fraction of the cluster's EM energy.

- E_{HAD}/E_{EM}: The ratio of the hadronic energy in the EM cluster to the amount of EM energy.

- E_T/p_T: The ratio of the transverse EM energy in the cluster to the COT's transverse momentum.

- L_{SHR}: The lateral shower profile for electrons. See [34] for details.

- $Q * \Delta x$: The distance in the r-ϕ plane between the COT track and the best matched CES cluster, times the charge of the track.

- Δz: The distance in the r-z plane between the COT track and the best matched CES cluster.

- χ^2_{strip}: The statistical χ^2 of the CES shower profile in r-z, compared to test beam data.

- z_0: The z position of the COT track along the beamline axis.

- Track quality: The COT track must possess segments through 3 axial super-layers and 3 stereo super-layers, with each super-layer having no less then 7 (of a possible 12) hits.

7.1.2 Muons

This section describes the cuts applied to define muons objects. Any COT track associated with a stub in one of the muon systems that also meets the following criteria is called a muon. This analysis only makes use of CDF's tight CMUP or CMX muons.

Quantity	Cut
E_T	> 20 GeV
E_{HAD}	$< 6.0 + \text{MAX}[0, 0.0280 \times (c\, p_T - 100)]$ GeV
E_{EM}	$< 2.0 + \text{MAX}[0, 0.0115 \times (c\, p_T - 100)]$ GeV
$\lvert\Delta x\rvert$ CMU	< 7 cm
$\lvert\Delta x\rvert$ CMP	< 5 cm
$\lvert\Delta x\rvert$ CMX	< 6 cm
$\lvert d_0\rvert$ silicon track	< 0.02 cm
$\lvert d_0\rvert$ non-silicon track	< 0.2 cm
Isolation Cut	< 0.1 in $R = 0.4$ cone
COT hits	≥ 7 hits on 3 Axial & 2 Stereo Super-Layers

Figure 7.4: CMUP/CMX muon cuts

- E_{HAD}: Hadronic calorimeter energy threshold.
- E_{EM}: EM calorimeter energy threshold.
- $|\Delta x|$: The magnitude of the r-ϕ distance between the fitted COT track path and the muon stub location (in centimeters).
- $|d_0|$: The magnitude of the impact parameter for the associated track.
- Isolation Cut: The ratio of the total p_T in a cone of radius $R = 0.4$ around the muon (excluding its own) to the muon's p_T.

Also, all muons that have a silicon track have the so-called "larry" correction [35] applied to their track curvature, such that:

$$\frac{1}{q \cdot p_T} \rightarrow \frac{1}{q \cdot p_T} - 0.00039 - 0.00129 \cdot \sin(\phi + 0.47) \tag{7.1}$$

Finally, any muon stubs used must pass a cosmic ray veto [36] to ensure that their timing characteristics are consistent with having originated from the detector's centre.

7.1.3 Jets

At CDF, a "tight" jet is defined to be a jet reconstructed using the JetClu-0.4 algorithm that possesses a minimum transverse energy of 20 GeV. The details of the JetClu algorithm can be found in CDF note 7543[32]. The 0.4 refers to the radius $R = \sqrt{\Delta\eta^2 + \Delta\phi^2}$ of the cone in η-ϕ space used to reconstruct the jet. As η increases, the jet cone becomes somewhat squashed along the z-axis. To illustrate the extent of a jet's size about the collision point, Figure 7.5 plots some example cones of radius $R = 0.4$ with centers located at increasing values of η.

In general terms, the algorithm computes the following quantities based on "seed" towers with large amounts of transverse energy.

$$E_T^{jet} = \sum_i^{N_{tow}} E_{Ti} \tag{7.2}$$

$$\phi^{jet} = \sum_i^{N_{tow}} \frac{\phi_i E_{Ti}}{E_T^{jet}} \tag{7.3}$$

$$\eta^{jet} = \sum_i^{N_{tow}} \frac{\eta_i E_{Ti}}{E_T^{jet}} \tag{7.4}$$

Each seed tower builds "clusters" composed of the towers located within a radius R that have $E_T > 1$ GeV.

This procedure is repeated iteratively by selecting a new list of towers located around the computed centers for ϕ and η in (7.3) and (7.4). This process continues until the center of the jet becomes stable. If any two jets overlap by more than 50%, they are combined into a single jet. If there is less than a 50% overlap, the offending towers are assigned to the jet with the closest center. The final jet parameters are then computed per the following definitions (using the final combination of towers):

$$E^{jet} = \sum_i^{N_{tow}} E_i \tag{7.5}$$

$$p_x^{jet} = \sum_i^{N_{tow}} E_i \sin\theta_i \cos\phi_i \tag{7.6}$$

$$p_y^{jet} = \sum_i^{N_{tow}} E_i \sin\theta_i \sin\phi_i \tag{7.7}$$

$$p_z^{jet} = \sum_i^{N_{tow}} E_i \cos\theta_i \tag{7.8}$$

$$p_T^{jet} = \sqrt{(p_x^{jet})^2 + (p_y^{jet})^2} \tag{7.9}$$

$$\phi^{jet} = \tan\left(\frac{p_y^{jet}}{p_x^{jet}}\right) \tag{7.10}$$

$$\sin\left(\theta^{jet}\right) = \frac{p_T^{jet}}{\sqrt{(p_x^{jet})^2 + (p_y^{jet})^2 + (p_z^{jet})^2}} \tag{7.11}$$

$$E_T^{jet} = E^{jet}\sin\left(\theta^{jet}\right) \tag{7.12}$$

Jets that have $E_T < 3$ GeV are typically not used. The jet energies also undergo a correction process to account for various detector and physical effects. The corrections are broken down into levels, of which there are seven (7):

- **Relative:** Normalize the calorimeter response to be uniform in η.
- **Time Dependant:** Corrects jets for any effects in the calorimeters due to time (*i.e.* effects that change with Run Number).
- **Energy Scale:** Corrects the jets for differences between Real Data and Monte Carlo.
- **Multiple Interactions:** Take account of any extra parton interactions from the $p\bar{p}$ collision.
- **Absolute:** Convert the jet's total cluster energy back to the particle level.
- **Underlying Event:** Subtract out the energy of the underlying event, to leave only the energy of the jet's originating parton.
- **Out-of-Cone:** Correct for any energy from the parent particle lost outside the cone radius R.

This analysis follows the lead of CDF's top quark working group, and corrects jets up to and including level 5. These corrections are applied to all jets (both MC and

RD) to make them align as best as possible. This will be discussed further in Section 9.4.1, which deals with the jet energy scale systematic error.

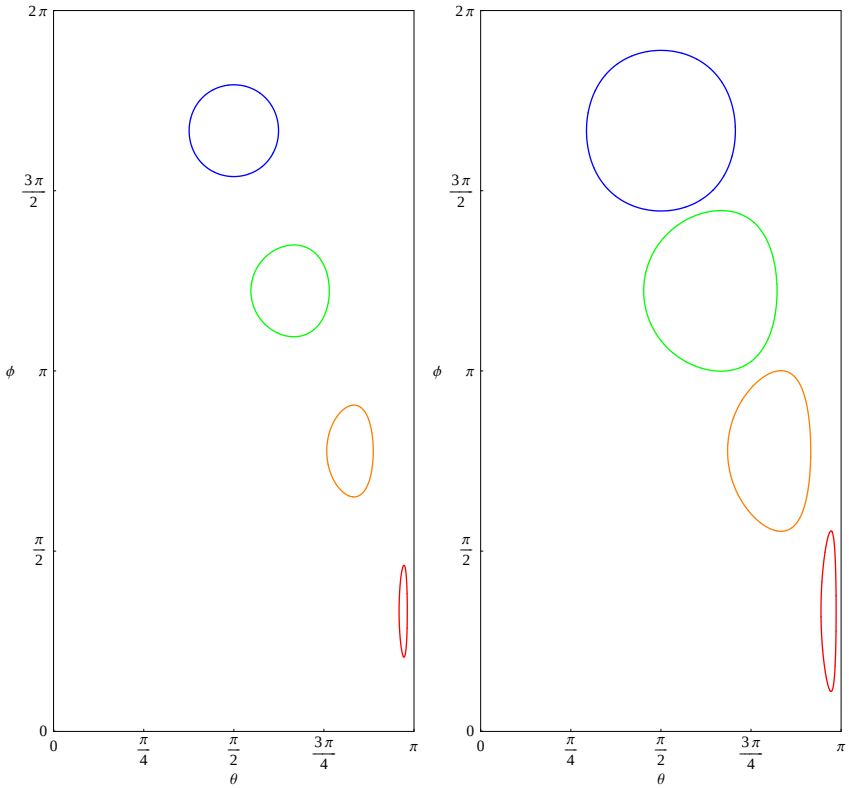


Figure 7.5: Jet cones of radius $R = 0.4$ (left) and $R = 0.7$ (right)

7.1.4 Missing E_T

The missing transverse energy ($\not{E}_T$) is specifically defined as follows:

$$\not{E}_{Tx} = -\sum_{i}^{\ell+j's} E_{Ti} \cos\phi_i \tag{7.13}$$

$$\not{E}_{Ty} = -\sum_{i}^{\ell+j's} E_{Ti} \sin\phi_i \tag{7.14}$$

$$\not{E}_T = \sqrt{(\not{E}_{Tx})^2 + (\not{E}_{Ty})^2} \tag{7.15}$$

$$\phi_{\not\!\!E_T} = \tan^{-1}\left(\frac{\not\!\!E_{Ty}}{\not\!\!E_{Tx}}\right) \tag{7.16}$$

where the sums in Equations (7.13) and (7.14) are over the lepton and jet objects in the event. Note that when applying jet corrections, the $\not\!\!E_T$ and its direction are recomputed to reflect any energy shifts. Also, since muons are minimally ionizing in the calorimeter, any that were identified are added in a special way. Their minimal energy deposits are subtracted out, and then their total reconstructed p_T is added to our vector sum.

7.2 Complete Event Selection Criteria

We are now in a position to enumerate the entire initial event selection criteria used in our $\sigma_{t\bar{t}}$ analysis. These requirements make use of CDF's latest lepton plus jets (L+J) selection standards.

For both Real Data and Monte Carlo, the selection that follows was used. We required exactly one (1) tight electron or one (1) tight muon. The event must have contained three (3) or more tight jets, and have had $\not\!\!E_T$ greater then 20 GeV. Further, we required that there be no plug (phoenix) electrons present, and that five (5) other criteria (described below) were met. In terms of pseudo-code, the requirements were as follows:

```
if(
(( nTightMuo == 1 && nTightEle == 0 ) || ( nTightMuo == 0 && nTightEle == 1 )) &&
( nTightJet >= 3 ) && ( corrMetMag > 20.0 ) && ( nPhoenix == 0 ) &&
!isQCD && !isDilepton && !isZ && !isDifferentInteraction && !isCosmic
)
```

where && is the logical AND, || the logical OR, == logical equivalence, and ! the logical NOT.

"isQCD" is true when the angle $\Delta\phi$ between the leading jet and $\not\!\!E_T$ is less then 0.5 or greater than 2.5 radians and the magnitude of the $\not\!\!E_T$ is less than 30 GeV. "isDilepton" is true when the summary bank from the top ntuple has two or more ($\geq$ 2) tight leptons recorded (plug electrons included). "isZ" and "isCosmic" also come from the top summary bank in the ntuple. They serve to veto events in the Z^0 boson mass window or events that don't pass the cosmic ray filter. "isDifferentInteraction" is

true when the z_0 of the tight muon or electron is greater than ± 5 cm from the event's z-vertex object.

The selection criteria was applied to pythia $t\bar{t}$ signal (ttop75), $W \rightarrow \ell\nu$ + lf backgrounds (ptopXX, utopXX), $W \rightarrow \ell\nu$ + bb backgrounds (btopXw, dtopXw), $W \rightarrow \ell\nu$+cc backgrounds (ctopXw, etopwX), and $W \rightarrow \ell\nu$ + c backgrounds (stopwX). All these MC datasets were used in the training of our SVMs. After training, dibosons, single top, and Z+jets samples were also considered for use within the background templates.

7.3 Features

In order to describe our physics events, we must decide on a collection of quantities which capture the character of the different classes in our problem. First, we would like to make sure we are using all the information available to us in the event. As was discussed above, the major players in the topology of the events we select are a tight, high transverse momentum lepton, three or more high p_t jets, and a large missing transverse energy. We can deal quickly with the $\not\!\!E_T$, by recording its magnitude and longitudinal angle ϕ. Since we have exactly one (1) lepton in the event, it makes sense to grab its 4-momentum as one of our features. This should fully describe that part of the event. Next, since there are potentially many jets, we need some description which will capture their overall distribution. There are an unknown number of 4-vectors that come from jets, so let's try several things to get as much out of them as is possible. The overall vector sum of the 4-momentum is a good first start. Also, we can take the scalar sum of the jet energies. However, another series of quantities, known as Fox-Wolfram Moments, can be used to quantify the angular distribution of the jets from an event. Fox-Wolfram Moments are defined as follows:

$$H_\ell = \frac{4\pi}{2\ell+1} \sum_{m=-\ell}^{\ell} | \sum_{i}^{\text{jets}} \frac{E_T\,(i)}{E_T\,(\text{Total})} Y_\ell^m\,(\theta,\phi)\,|^2 \tag{7.17}$$

As can be seen, they are the spherical harmonics [$Y_\ell^m(\theta,\phi)$'s], fractionally weighted by jet transverse energy. They are constructed such that each moment ℓ corresponds to the basis functions, which form an orthonormal basis about the unit sphere. Recall that $Y_0^0 = \frac{1}{2\sqrt{\pi}}$, so that $H_0 = 1$, a constant, by definition. Things only get interesting for $\ell > 0$. We will keep the first five moments in our ntuples. One nice property of

these Fox-Wolfram moments is that they are invariant under any global rotation in θ or ϕ.

Now we might like to know how the transverse quantities correlate. In a scalar fashion, we can construct the so-called H_T. It is defined as the total of all the transverse quantities in the event. In other words, it's simply the sum total of the jet's E_T, $\not\!\!E_T$, and lepton E_T. These quantities can also be combined, retaining some of their vector information, in a normalized momentum tensor.

The normalized momentum tensor is given as:

$$P_{ij} = \frac{\not\!\!E_{Ti}\not\!\!E_{Tj}}{\left|\not\!\!E_T\right|^2} + \sum^{\ell+5\ jets} \frac{p_i p_j}{|p|^2} \tag{7.18}$$

where i and j run over xyz only. For the jets, we only take as many as five, those with the highest E_T. We will calculate this tensor for each event, and then keep its two largest eigenvalues.

An additional factor which can help in describing our events is the fact that we expect there to be a symmetry in ϕ, the longitudinal angle about the beamline of the accelerator. As such, we can reduce the "phase space" of events we select by rotating each event in ϕ such that the lepton in the event travels along $\phi = 0$, or parallel to the ground in the coordinate system of the CDF detector.

Describing our events with these features gets a good handle on the kinematic aspects of the decay products. However, another piece of information at our disposal is the fact that the top quark almost always (very quickly) decays to a bottom quark and W boson ($t \to bW^+$, $\bar{t} \to \bar{b}W^-$). Therefore, the decays we are trying to identify should contain two jets originating from b quarks. There are methods which can be used to identify jets which come from bottom quarks, which is known in particle physics lingo as b tagging the jets. A jet which has been b-tagged is then considered to be a jet which resulted as a consequence of the hadronization of a bottom quark. A simple count of the number of jets within the event that are tagged should then give us some discriminatory power. CDF has several methods which are used to tag jets. The one we will employ in this analysis is the Secondary Vertex Method (SecVtx). A brief description of this method follows below. Some of the other methods available are jet probability and the soft lepton tagger. See the following reference [4] for more information about these tagging methods.

The basic principle used in the SecVtx b tagging method is to look for another (*i.e.* secondary) vertex displaced from the primary vertex in the event. The primary vertex

represents the main interaction point of the $p\bar{p}$ collision. Any secondary vertex located an appropriate length away from the main collision, as to correspond with the flight time of a bottom quark over its lifetime, can be tagged.

In summary, we create vectors with twenty (20) dimensions in order to characterize the events we have saved after the application of the event selection described in the earlier parts of this chapter. Each of these quantities is listed in Figure 7.6.

Note that one final step is carried out before these values are fed into the SVM framework. So that the features each carry something of the same weight, and each are considered on an equal footing in the optimization process, the magnitudes of the individual features are scaled to approximate unity. The number used to divide each feature is recorded in Figure 7.6 as well. These specific values were found empirically by the examining the range of values each variable exhibited.

Num	Feature	Scaled by
1	Missing E_T	250
2	Missing E_T ϕ	2π
3	Lepton Energy	400
4	Lepton p_x	400
5	Lepton p_y	400
6	Lepton p_z	400
7	Fox-Wolfram Moment 1	1
8	Fox-Wolfram Moment 2	1
9	Fox-Wolfram Moment 3	1
10	Fox-Wolfram Moment 4	1
11	Fox-Wolfram Moment 5	1
12	Sum All Jets E_T	500
13	Sum All Jets Energy	750
14	Sum All Jets p_x	750
15	Sum All Jets p_y	750
16	Sum All Jets p_z	750
17	H_T	500
18	p-Tensor Eigenvalue 1	1
19	p-Tensor Eigenvalue 2	1
20	Number SecVtx Tags	10

Figure 7.6: Feature Vectors

Chapter 8

SVM Training

Following our analysis outline, we now know how to decide if we want to keep any given event for further study, and how to create a vector of features to describe that event. The next step is to take simulated events from known underlying processes, and use them to train Support Vector Machines that will be able to tell those processes apart from $t\bar{t}$ pair production.

In order to train our SVMs, the following procedure was followed. The Monte Carlo sample was split into two sections. One set was used to train the machine. The remaining events were then used to test the training's performance. Also, it was these remaining events that were used in any subsequent pseudo-experiments and analysis. Training events which passed our initial selection cuts were randomly picked from each Monte Carlo sample. Six thousand (6,000) $t\bar{t}$ events for the Signal class [S], and twelve thousand (12,000) background events (6k for the Heavy class [H], and 6k for the Light class [L]) were used to train SVMs with a Gaussian kernel. As was described in the theory section of this document, this means that there are two free parameters that needed to be specified when training. These are the σ parameter of the Gaussian kernel, and C that constrains the Lagrange multipliers in the solution. A search grid was implemented over these parameters, and the performance of a given training was then evaluated based upon how well a testing sample was classified. As an example of the results from this kind of grid search, see Figure 8.1. It shows classification performance as a function of σ and C. Because the light class is the more dominant background, we chose the hyper-parameters to maximize the performance of the SvL SVM. In order to maintain the same feature space across all the SVMs in the system we created, σ and C were always taken to be the same for each pairing of classes. This

allows us to use the Gram-Schmidt procedure described earlier in Section 3.3.

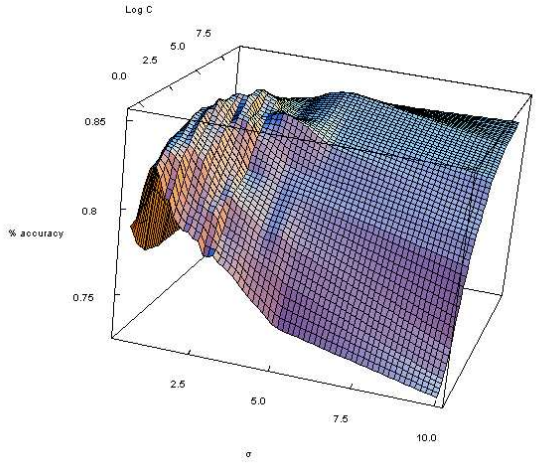


Figure 8.1: Sample Training Grid Search

In our initial trials, two different SVM systems were trained with the Monte Carlo data. One used only the kinematic features, which includes all the features listed in Figure 7.6 except for the last one. This SVM set was used to learn and perfect our methodology, and also as a baseline to compare against our future discriminatory gains. A second set of SVMs was trained to included all features (adding the SecVtx b-tag feature), and this 2^{nd} set was the one ultimately employed to measure $\sigma_{t\bar{t}}$.

It is worth noting that there were attempts made at using more features, some of which included information from more than one b-tagging scheme. However, because of difficulties in trying to correct each of the different b-tagging features to align with Real Data (RD), it was decided to only include the number of SecVtx tags to supplement the kinematic quantities. This allowed us to correct events by the SecVtx b-tagging scale factor between MC and RD by dropping some b-tags in the MC as appropriate. See the systematic error Section 9.4.5 later in this document for a more detailed description of this procedure.

We will refer to the two trained SVM systems as the kinematic version and the b-tagging version, respectfully. Their overall performance when using the sign of the learned function for classification is listed in Figure 8.2. The final values that were chosen for the hyper-parameters are also listed there.

Here, "FP%" means false positive rate, and represents the percentage of the time the classifier misidentifies signal events. Similarly, "FN%" stands for false negative,

SVM	MC Version	σ	C	FP%	FN%
kinematic	1invfb	1.2	10	20.0	18.0
b-tagging	1invfb	1.2	10	11.0	15.0

Figure 8.2: Training Results

and tells us how often the specified SVM mislabels background events.

8.1 Receiver Operator Characteristic

Another performance metric is given by the so-called Receiver Operator Characteristic (ROC) curve. This plot shows how the performance of a SVM varies with the bias (b) of the learned function. By construction, the bias (b) computed in the training process gives the optimal combination of false positives ($FP\%$) and false negatives ($FN\%$) for the selected hyper-plane. However, if the bias is slid from $-\infty$ to $+\infty$, gains can be made in one category at the expense of the other.

The ROC curve is a standard statistical plot that illustrates this interrelationship for a given classification scheme. Traditionally, it plots the $FP\%$ on the x-axis, and $(1 - FN\%)$ on the y-axis. The area under the generated curve can be used to compare different classifiers. A classifier that is no better then a coin produces an ROC curve with area 0.5, while a perfect classifier has unit area.

Figure 8.3 shows ROC curves for both the kinematic and b-tagging SVMs which were trained with the class pairing ($t\bar{t}$ signal *vs* light background). They have areas of 0.871 and 0.941, respectivly. Another series of ROC curves are plotted in Figure 8.4. These compare the performances of the three SVMs trained in our multiclass framework.

It can be shown that in the limit of a perfectly crafted ROC curve (*i.e.* the PDFs are known exactly), its area represents the probability of correctly classifying a randomly chosen pair of events, one from each class. In other words, if you were to pick a random signal event and a random background event, the area under the ROC curve is the probability that the learned function value of the signal event would be greater than that of the background event.

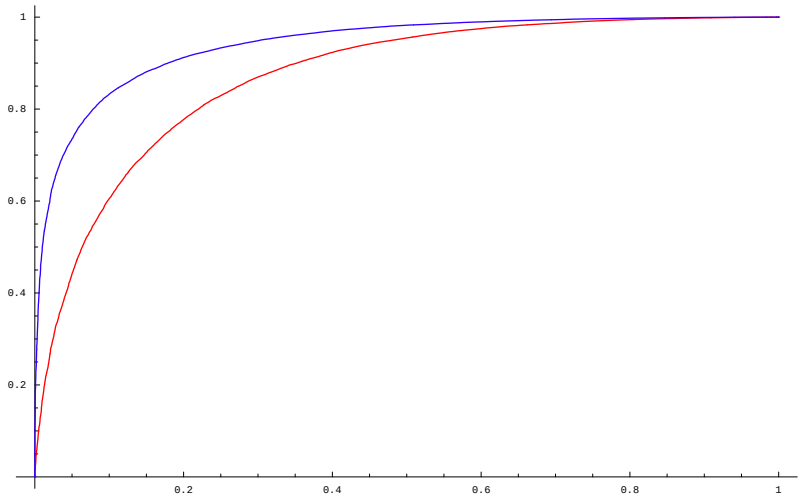


Figure 8.3: ROC Curves for kinematic and *b*-tagging SVMs

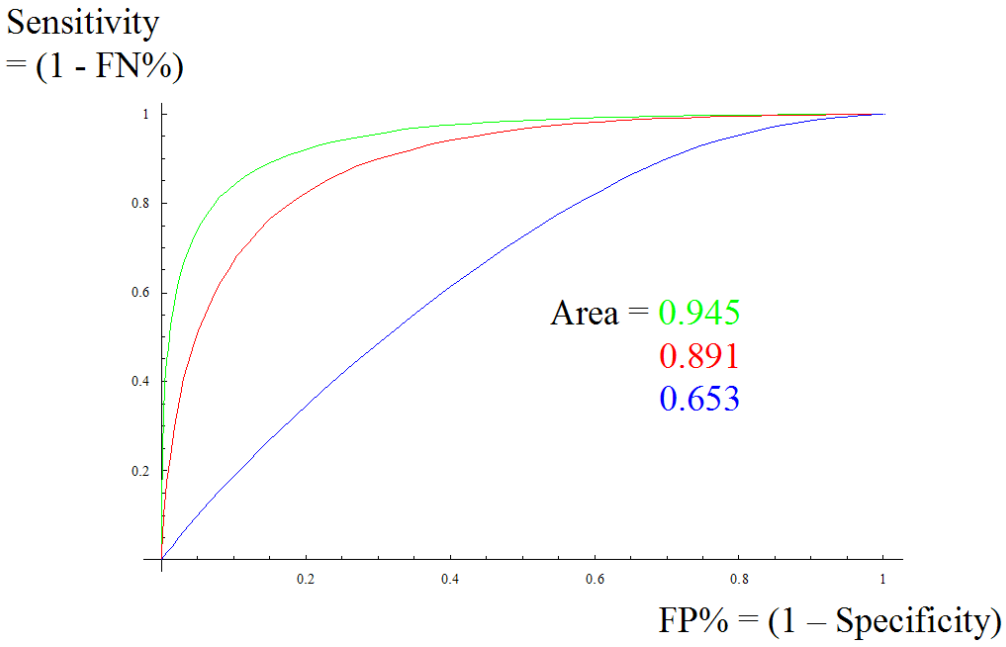


Figure 8.4: ROC Curves for SvL, SvH, and LvH SVMs

Chapter 9

SVM Performance & Error Analysis

9.1 Counting Events

9.1.1 PDFs & Probabilities

After deciding on the optimal values for σ and C, it is instructive to histogram the values of the learned function for each of the events in the test set. The SVM output for the $t\bar{t}$ test events from the b-tagging SVMs are displayed in Figure 9.1. These distributions represent the response of the SVM to a wide range of signal and background events. By treating them as probability density functions, we can construct the probability for a given event to be signal or background with respect to that SVM's training data. For example,

$$P_{sig}(\vec{x}) = \frac{PDF_{sig}(\vec{x})}{PDF_{sig}(\vec{x}) + PDF_{bg}(\vec{x})} \tag{9.1}$$

The PDFs are important because we have used them to characterize the data. The PDF curves as shown would be sufficient to employ the Pairwise Coupling scheme of Section 3.2. However, in the multi-class method of multiple SVMs we have employed here, it was seen that the full feature space distributions using the procedures of Section 3.3 were superior. Thus, we used 3D histograms of points calculated via Equations (3.8), (3.9), and (3.10) to characterize events. The following sections will flesh out this process.

Note that to be useful in our calculations below, we did some processing to turn any

binned histograms into continuous functions. The bin centers were linearly interpolated to create continuous functions that follow a curve. These interpolated functions were then used in any subsequent calculations.

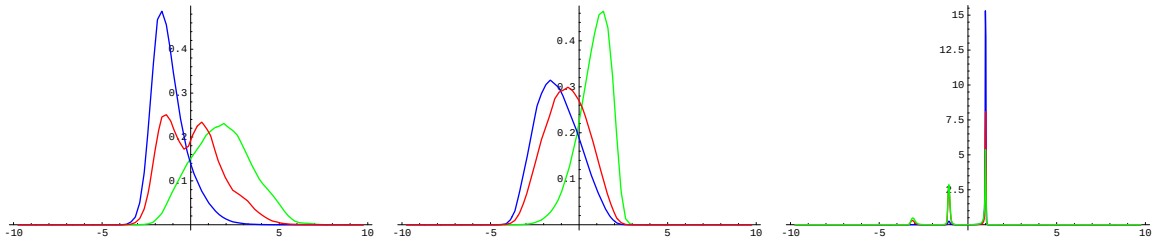


Figure 9.1: PDF templates for individual SVMs. (SvL,SvH,LvH)

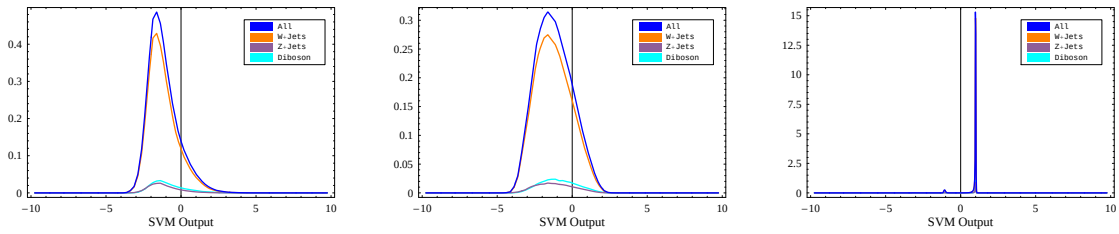


Figure 9.2: Light BG Template Breakdown

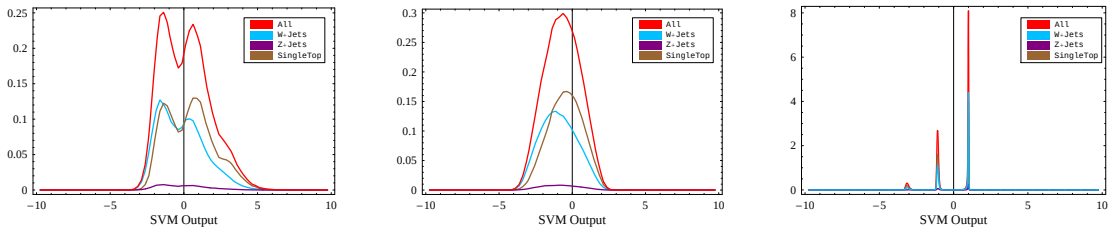


Figure 9.3: Heavy BG Template Breakdown

9.1.2 Likelihoods

Ultimately, we are interested in determining the relative amounts of signal and background in an unidentified sample. Probability gives us a measure of how likely a particular outcome is before an event takes place. However, we have set ourself the

opposite task. We are given a number of events that have already taken place, and would like to know how much of one outcome occurred as opposed to another. To do this, we employ the idea of likelihood. When trying to count the actual number of signal events in a given blind sample, the following procedure was followed.

$$L = \prod_i (\theta p_i + (1-\theta) n_i) \tag{9.2}$$

$$\Lambda \equiv \ln(L) \tag{9.3}$$

We extend these equations to the three class problem as:

$$L = \prod_i (\theta S_i + \phi L_i + (1-\theta-\phi) H_i) \tag{9.4}$$

$$\Lambda = \sum_i \ln(\theta S_i + \phi L_i + (1-\theta-\phi) H_i) \tag{9.5}$$

Assuming we are given an unknown sample of size N, we can pass each of these N event vectors through a trained SVM. Let us define θ to be the fractional amount of signal in the unknown sample, and is the value we would like to determine. The likelihood for an event is taken as the value of the probability distribution function at that event's value of the learned function. As in Equation (9.2), this will be some weighted combination of the signal and background PDFs (*i.e.* templates). We have an approximation of these PDFs from the distribution of events given by the learned function of the trained SVM. The overall likelihood of the ensemble is just the product of the likelihoods for each of the N events. The likelihood function can then be extremized with respect to θ. The value of θ at the extremum is our best guess at the amount of signal in our sample.

As a matter of computability, usually the logarithm of the likelihood function is taken. This is because samples usually consist of a large number of events, and products with a large number of terms can quickly grow out of hand. Taking the logarithm turns the product into a sum, has the same extremum as the product, and keeps the problem tractable.

As you can see, the likelihoods we will use are computed in terms of the probability distribution functions in the previous section. For example, we can define $p_i \equiv PDF_{sig}(\vec{x_i})$ and $n_i \equiv PDF_{bg}(\vec{x_i})$. A Monte Carlo sample separate from the

training sample is used to generate the PDF templates. Once the PDFs are known, we have fully defined a method for determining θ in an unknown sample.

Another way to view this procedure is the rescaling of the interpolated PDF functions calculated in the above section. A histogram of the SVM learned function over the blind sample should follow the distribution of the PDF functions if their relative areas are renormalized to the ratio $\theta/(1-\theta)$.

When extending our procedures to include more then 2 class templates, the likelihood function can be extended as in Equation (9.4). Here, S_i, L_i, and H_i are generally PDFs, or specifically the values given by our 3D templates from SVM feature space. θ and ϕ are the fractional amounts of the signal and light classes, and the fractional amount of the heavy class has been constrained to be $(1-\theta-\phi)$.

Another way to think of the likelihood for an event is to assume we have a large bag of marbles, each with an internal core which is colored to represent its class. The only way to see the core's color is to break open the marble. On the marble's surface are features that otherwise describe things about the properties of that particular sphere. We would like to guess the number of each type of marble in our bag, based only on the observation of these external features. Then, the likelihood of any specific marble we pick from the bag is given by the prior probability of picking a certain color (*i.e.* the fraction of green marbles in the bag) while at the same time having a marble of that color with this one's external features (*i.e.* the probability our external features are an expression of green-ness). If we sum this product for the marble over all its possible colors, the resulting quantity gives a measure of how likely we were to pick such a marble out of the bag. If we then compute this likelihood for every marble we select, their overall product tells how likely we were to assemble that collection of marbles. By maximizing this likelihood with respect to the prior fractional amounts of each color in our bag (*e.g.* θ and ϕ), we can estimate the bag's overall composition. Obviously, the more marbles we are able to examine from the bag, the better our guess becomes as to its content.

In order to maximize our Equation (9.5), we used the C++ version of MINUIT. Because MINUIT is not designed to impose mutual constraints on the parameters of optimization (here that $0 \leq \theta \leq 1$ and $0 \leq \phi \leq 1$ and $\theta + \phi \leq 1$), a transformation of variables was used to impose this restriction. See Appendix D for the details of this transformation, and its implementation. As a cross check on the answers generated by MINUIT with this change of variables, the results obtained during development were verified using a (much slower) grid search over θ and ϕ. Figure 9.4 shows a contour

plot of the log-likelihood Λ for a typical pseudo-experiment, and also the analogous plot when computing Λ for the Real Data.

9.2 Pseudo-Experiment Procedure

In order to evaluate various aspects of the SVM system we have trained, a series of pseudo-experiments were performed. A pseudo-experiment is an attempt to recreate and simulate the circumstances of the measurement we are making. Running many of these pseudo-experiments gives insight into how we can expect the results of our actual experiment to change, allowing for modifications of certain assumptions made in the construction of our mathematical apparatus. For instance, we chose certain mathematical Monte Carlo generators to simulate the $t\bar{t}$ (and other physical processes). We can do pseudo-experiments using different generators to examine how this would affect the results of the cross section measurement. Changing the various knobs associated with underlying assumptions used in training our SVMs gives us a handle on systematic effects which may arise from those initial choices. However, we must first have a baseline performance with which to compare these changes. This baseline will also be evaluated in terms of pseudo-experiments, but using the statistical nature of the physics involved in the events we are examining.

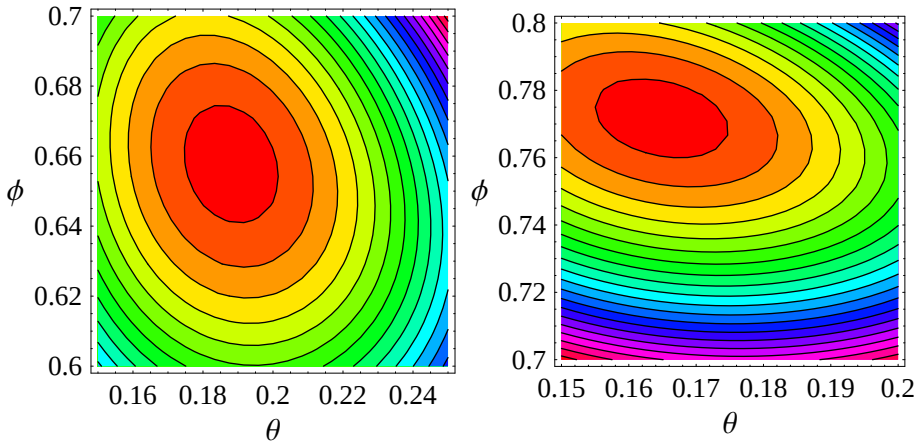


Figure 9.4: Contour Plots for Pseudo-Experiment (left) and Real Data (right)

9.2.1 Class Composition

Each pseudo-experiment has a basic premise. We will draw a certain number of events from each of the various classes which were used to construct our class templates. The total number of events selected will be Poisson fluctuated around a mean value M, chosen to be the actual number of real data events that pass our initial selection criteria (see Section 7). The number of events from each class will be chosen in an appropriate way to test various aspects of the SVM system. The first thing we will examine is the "linearity" of the system we have trained. By this, it is meant that we would like to see if the SVM procedure we have devised gets the correct answers when examining mixtures of the 3 classes ($t\bar{t}$ signal, light BG, heavy BG). Doing so will expose any biases within the framework of our procedure.

$$M = \mu_S + \mu_L + \mu_H \tag{9.6}$$

It is expected that the heavy background will account for between 5% to 10% of the overall W+Jets background. For what follows we will assume a mixture of 7.5% Heavy events to 92.5% Light events. As was discussed in the training section of this report, the various sub-processes within the overarching Heavy and Light delineations are determined by their expected cross sections within that class.

To serve as an example, the following paragraphs will describe in detail the composition of an example Heavy class, built up from random selections taken from its sub-processes. An analogous procedure is used to construct the Light class. To ease the discussion here, we will assume that the Heavy class is made up of nine (9) separate sub-processes, which are all the permutations of W(e,μ,τ)+bb+(0,1,2)p. Table 9.5 shows the expected cross sections for each of these sub-processes. Note that the actual Heavy class used in our full measurement also includes Z+bb and single top events (as is discussed later in this document). With the knowledge that we would like to construct a sample of size with Poisson fluctuated mean μ_H, we will again utilize the fact that a mixture of samples taken from several Poisson distributions is itself Poisson distributed (see Appendix B). Translating the cross sections in Table 9.5 into the relative fractional representation of each process yields Table 9.5's rightmost column. These values are calculated by weighting the number of events from a process that pass the selection cuts by its cross section, and then renormalizing these values to sum to unity.

For example, the process We+bb+1p had generated for it 1,459,427 events, of

Dataset	Process	cross section	pre-selection fraction	raw accept	post-selection fraction
btop0w	We+bb+0p	2.980 pb	23.9%	0.067%	2.962%
btop1w	We+bb+1p	0.888 pb	7.1%	1.159%	15.264%
btop2w	We+bb+2p	0.287 pb	2.3%	7.559%	32.167%
btop5w	Wμ+bb+0p	2.980 pb	23.9%	0.057%	2.522%
btop6w	Wμ+bb+1p	0.889 pb	7.1%	1.015%	13.383%
btop7w	Wμ+bb+2p	0.286 pb	2.3%	6.571%	27.866%
dtop0w	Wτ+bb+0p	2.980 pb	23.9%	0.008%	0.334%
dtop1w	Wτ+bb+1p	0.888 pb	7.1%	0.134%	1.769%
dtop2w	Wτ+bb+2p	0.286 pb	2.3%	0.880%	3.734%

Figure 9.5: Heavy Class ($W \rightarrow \ell\nu$+bb+Jets) composition

which all were potentially CEM / CMUP events and 1,443,652 could have been CMX events. Of these, 16,918 CEM events were reconstructed (zero CMUP and zero CMX). Therefore, this process has a raw acceptance of 16,918 / 1,459,427 = 0.0116 or 1.16% in the CDF detector. Given the cross sections in Table 9.5, a perfect detector would expect to see $\sigma_{bb+1p}/\sum_i^{AllW+bb}\sigma_i$ = 0.888/12.464 = 0.07125 or 7.125% of all W+bb events to come from We+bb+1p. To correct this amount for the detector effects, the product of the process's raw detector acceptance and theoretical expected fraction is taken. This is then renormalized by the sum of these products over all the sub-processes in the class, and thus gives the corrected post event selection fractional composition for that sub-process. For We+bb+1p, this turned out to be 15.3%.

9.2.2 Template Construction

Once we know the frequency at which we can expect to find each class's sub-processes to appear, we are able to build templates to represent the class's reaction to the SVM system we have trained. The templates for each class are created in the following way. Each event that was not used in the training process is classified by each of our three SVMs. This triplet is then transformed via the Gram-Schmidt process described in Section 3.3. This provides us with a 3 dimensional point for each event, which is located in the sub-space of feature space that is picked out by the three hyperplanes we have trained. This three dimensional space is then used to build a function. To do this, we create a tri-variate histogram over the space. This simply means that our space is divided up into cubes of some size, and the number of events from our sample

which fall inside each cube is tallied. However, the tally we make is a weighted one. The event weights we calculated, as in the rightmost column of Table 9.5, are entered for each event depending upon the source sub-process of that event. By dividing this weight by the number of events we enter into the histogram from its sub-process, we automatically give that class an overall weight of the correct proportion. To be totally clear, again using the Heavy class as an example, the histogram we create over feature space would have We+bb+1p events entered with a weight of 0.153 / (number of We+bb+1p events inserted), and Wμ+bb+2p events entered with a weight of 0.279 / (number of Wμ+bb+2p events). Using this weighting procedure allows us to make maximal use of all the Monte Carlo events at our disposal for the construction of the template.

In order to create a continuous function from the histogram weight counts, the value of the function between adjacent cubes is linearly interpolated. Also, the overall size of the function is rescaled to have an integral area of one. For a histogram cube size of length dx, this is achieved by dividing the whole function by dx^3. This makes the template function somewhat analogous to a probability density function (PDF). Templates PDFs S, L, and H were created in this manner, and are functions of feature space in the following way (*e.g.*):

$$S(X(\vec{z}), Y(\vec{z}), Z(\vec{z})) \Rightarrow S(\vec{z}) \tag{9.7}$$

as the X, Y, and Z of Equations (3.8) - (3.10) are ultimately functions of test feature vectors $\vec{z}$.

The process of maximizing the likelihood function was carried out using MINUIT. To apply the constraint that $\theta + \phi + \gamma = 1$, a transformation of variables was used as described in Appendix D.

9.2.3 Template Results

Using the procedure of the previous section, templates were constructed for each of the three training classes using all the events available which were not used in training our SVMs. Some figures have been included to show these functions. Since three dimensions can be difficult to visualize, we have presented the resulting template (density) functions in two different ways. In Figure 9.6, the picture on the left shows a random sampling of the template functions, treating them as probability densities. A sampling of five thousand (5,000) points from each class was used. The picture on the right shows

the contour of the functions for the value 1,000. This value was chosen empirically to show the general behavior of the templates. Figure 3.2 shows some views of the feature space along with the three separating hyper-planes determined from our training. This picture illustrates nicely the effect of using the Gram-Schmidt procedure to place points in feature space, and how these hyper-planes form the basis for separating the three classes. The structure seen in these pictures can be understood in terms of the discrete nature of the SecVtx b-tag feature used in training. The three bands seen in the template graphs correspond to the groups of events with zero, one, or two btags. As expected, the blue Light class is mostly located in the zero band. Also notice the Light vs Heavy hyper-plane is located between events with zero tags, and those with more. This is also as one might naïvely expect.

As was discussed above, its possible to compute the gradient functions for each of the SVMs we have trained. This has been done, and then each gradient sampled with several $t\bar{t}$ events. The average gradient vector over these events is shown in Figure 9.7. As can be expected, b-tagging plays a large role in separating $t\bar{t}$ events from Light events, while the separation of $t\bar{t}$ from Heavy is much more kinematically oriented. Also in accord with our intuition, the Light class is distinguished from the Heavy almost entirely by the b-tag feature.

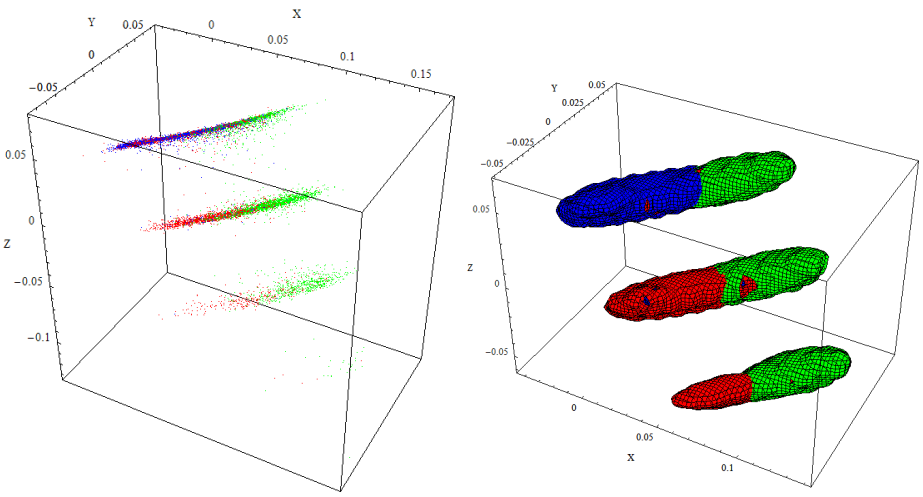


Figure 9.6: Template Plots (Signal, Light BG, Heavy BG)

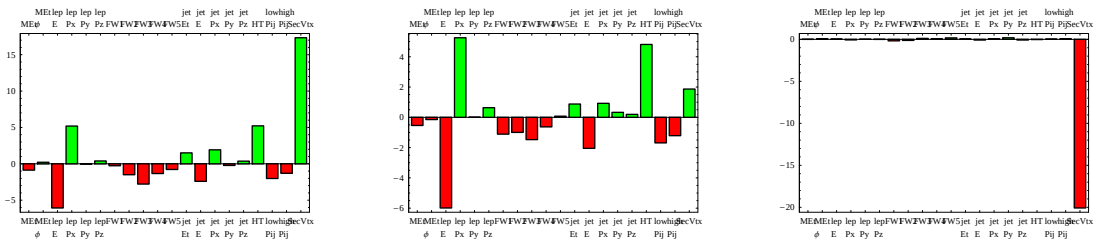


Figure 9.7: Gradients for (S vs L), (S vs H), and (L vs H) SVMs

9.2.4 Linearity

Pulling everything together, we will now examine the linearity of our multi-class SVM technique. Because we observe 6,888 events in the real data that pass our selection criteria, we choose this to be our mean events size M in our pseudo-experiments (PEs). As was also stated previously, we will fix the relative Light and Heavy proportions at 92.5% and 7.5% (or $12\frac{1}{3}$ Light events to each Heavy). Next we will take the mean $t\bar{t}$ fraction f_S to range from 5% to 95% in increments of 5% (or sometimes from 10% to 20% in 1% increments). Therefore,

$$\begin{aligned} \mu_S &= M f_S \\ \mu_L &= M(1 - f_S)(0.925) \\ \mu_H &= M(1 - f_S)(0.075) \end{aligned} \tag{9.8}$$

These three values were used to draw pseudo-experiments. Five hundred (500) PEs were drawn at each of the values of f_S. For each individual PE, the actual number of events for each class was determined by consulting Poisson distributions with means as given by μ_S, μ_L, and μ_H. This allows the specific class contribution to fluctuate from PE to PE. Note that for the composite Light and Heavy classes, the underlying sub-processes are also Poisson fluctuated in the manner described above in section (9.2.1). Once we have determined the specific number of events from each class to include, the corresponding number of events in random permutation from our Monte Carlo is chosen. This leaves us with a grouping of events that we will use to represent the actual measurement we will eventually carry out on the real data from the accelerator.

The values of the three template functions (S, L, and H) we have created are evaluated for each event in the pseudo-experiment. These values are used to create the

likelihood function, as is shown in Equation (9.4). This function is then extremized with respect to the mixture parameters (θ and ϕ) to get our estimate of the class compositions. The calculated θ and ϕ are compared to their actual values which were used to create the pseudo-experiment. The differences between these calculated values and the actual class composition is shown in Figure 9.8. The upper left plot in the figure shows the mean value of the absolute difference between θ and its expected value (i.e. $\Delta_\theta = \theta_{calc} - \theta_{true}$), as f_S moves from 10% to 20%. The other two top graphs show the coresponding plots for the Light and Heavy classes (ϕ and $\gamma = 1 - \theta - \phi$). The bottom of Figure 9.8 shows the standard deviation of these Δ's.

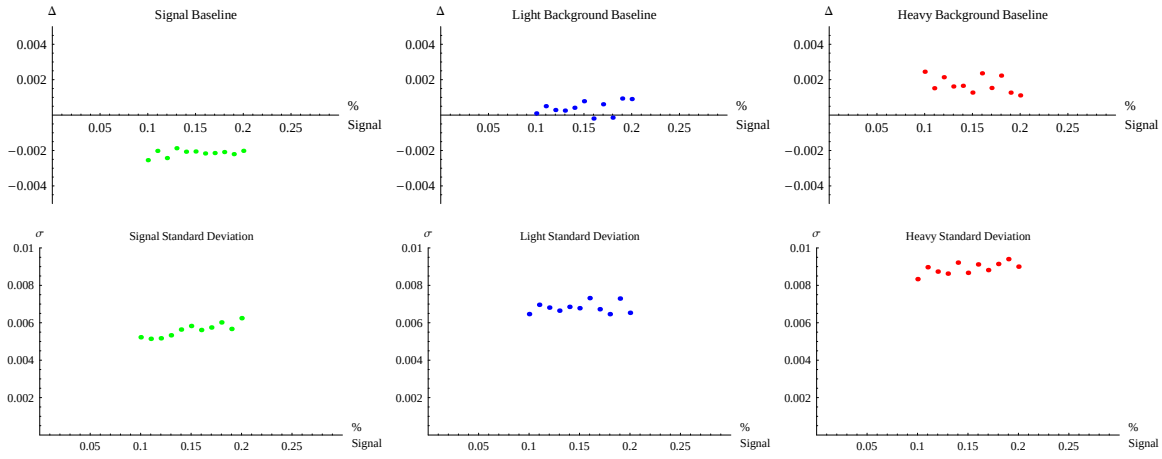


Figure 9.8: Baseline Linearity Plots. Mean Δ is shown on top and the Standard Deviation σ below

As can be seen from the plots, the Δ error in θ is fairly small and on the low side. The Light fraction Δ_ϕ is typically high, as is the Heavy class Δ_γ. The Light class has the smallest bias, but even at their largest these biases are constant and only -0.002 for Signal and +0.002 for Heavy. Ideally these plots would be completely flat, though some bias has to be expected because our training samples didn't include all the processes used in our templates (*i.e.* diboson, single top, or Z+Jets). It is difficult to completely understand the sources of these (albeit small) biases. They could also come from a statistical skew of the training data we used, or even the choices of binning and interpolation used in creating the template functions. Because they are so small, and can be mapped out with PEs, the approach taken will be to correct answers by these biases, treating them as a baseline correction.

The standard deviations shown give us an estimate of the statistical error in our

framework. To illustrate this, we have plotted in Figure 9.9 the relationship between this standard deviation and the mean number of events in the pseudo-experiment M. These PEs were performed at a fixed $f_S = 20\%$.

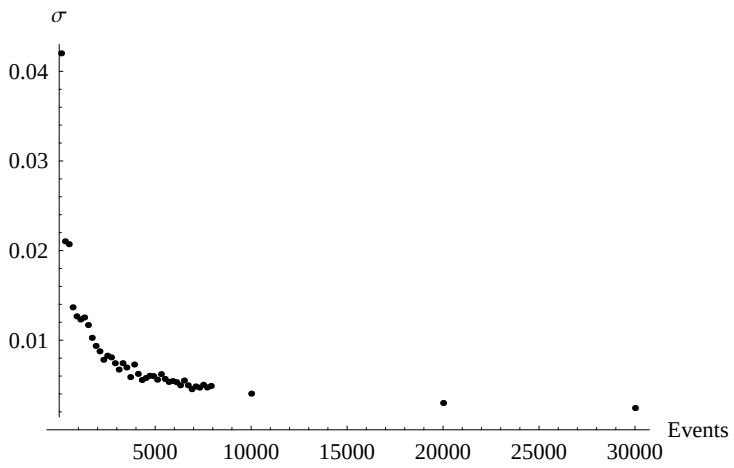


Figure 9.9: θ's statistical error as a function of the number of events in the pseudo-experiment

Also, to show the power of including the b-tagging information, this linearity test was also performed on the kinematic only SVMs. The results can be seen in Figure 9.10. The biases between the two formulations follow a similar trend. However, the standard deviation for the kinematic SVMs across all three classes is typically very much higher. Because we have no b-tagging information in the kinematic only formulation, one would expect our discriminatory power between the Heavy and Light classes to be greatly diminished. This is reflected in the large biases (sometimes 10%) with absolute standard deviations between 6% and 9%. Note that the bias in the signal class remains relatively small (at worst only about 0.5%), but that the standard deviation in $t\bar{t}$ here is roughly double that of the SVMs with SecVtx information included.

9.3 Other Backgrounds

There are a few other background processes which can mimic the signature of a $t\bar{t}$ decay. One kind of events so far unconsidered are the so-called non-W QCD fakes. This is a somewhat catchall group, that includes any QCD backgrounds which can fake a tight lepton in our detector. Other processes which have small cross sections are Z + Jets,

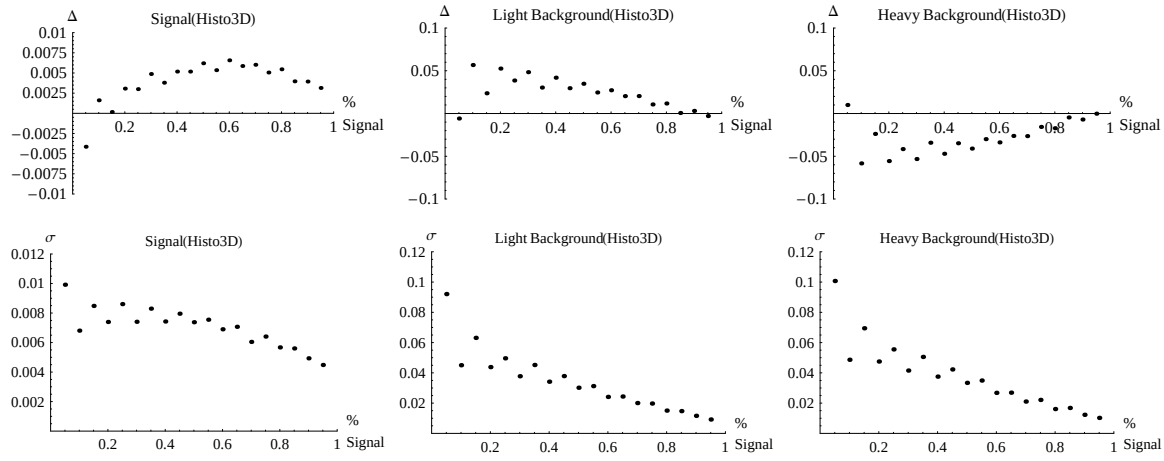


Figure 9.10: Kinematic SVMs' Linearity Plots

and the dibosons (WW, WZ, and ZZ + Jets). Finally, single top (*i.e.* the non pair production of a t or $\bar{t}$) can also be a source of background in this measurement.

9.3.1 Diboson

Events that contain two bosons (WW, WZ, or ZZ) can also sometimes give the same signature as $t\bar{t}$. CDF has Monte Carlo to represent these events. Datasets itopww, itopwz, and itopzz are combined by their expected cross sections, which are 7.70 pb, 2.30 pb, and 2.56 pb respectfully. This translates into a makeup of 83.4% WW, 14.2% WZ, and 2.4% ZZ for this analysis, after taking into account our event selection and the detector acceptances for these processes. These Monte Carlo datasets were added into the Light BG template, weighting these events by their cross sections relative to the other processes in this class. See Figure C.3 for a summary of the Light BG template and the processes included.

9.3.2 Z+Jets

As was noted above in Section 7.2, we use a Z veto in our initial selection criteria. This process removes events reconstructed with a Z boson found near its mass window. Consequently, the overall contribution from Z+Jets is small. However, to take their contribution into account, Z+Jets events were included in our two background templates. Z+lf and Z+cc events were folded into the Light Class, while Z+bb was

included in the Heavy Class.

9.3.3 Single Top

Single top production can mimic pair production in the case where one of the pair's top quarks is lost (*e.g.* down the beamline). Single top production occurs through two methods. These are the s-channel (where an intermediate W decays into a top & anti-bottom), and t-channel (in which a bottom quark transforms into a top through the exchange of a W with a up or down quark). Each has a relatively low cross section (0.88 pb and 1.98 pb respectfully). Because top quarks decay primarily as a Wb system, we have included the single top event signatures in the Heavy template.

9.3.4 non-W QCD Fakes

CDF employs two primary methods to model the QCD fakes. These are the jet-electron model, and the anti-electron model. Both use real data from the detector, and relax various aspects of the definition of a reconstructed electron. For jet-electrons, events which have a jet that is improperly identified as a tight electron are considered. Drawing on the pool of generic jet20 events, one looks for instances where a jet has E_T > 20 GeV, EMfraction between 0.80 and 0.95, and 4 or more tracks. Once identified, a jet-electron is treated as an electron and the rest of the event is subjected to all the rest of our selection cuts.

On the other hand, anti-electrons are objects very close to passing all the quality cuts assigned to the CDF definition of an electron. In order to qualify, two of the following five kinematic criteria must fail: CES Δx, CES Δz, Lshr, track χ^2, and hadronic fraction.

Because there are only some few hundred events that meet this definition in all the CDF data, they are not very useful in this SVM framework. Since our templates are based on trivariate histograms, it takes a good number of events to get a reliable function that describes the class in question. In contrast to the anti-electrons, jet-electrons are plentiful. They come from the generic jet datasets, a small sample of which produced a pool of some 35k events. This amount is entirely sufficient to create a template and run pseudo-experiments.

Two approaches were taken to evaluate the contribution of non-W QCD fakes to number of events which survive our selection criteria, and further to examine how they would be classified in our SVM framework.

As a first step, jet-electron events were used to construct a template in the feature space of our problem. Again, this means that all the jet-electron events were classified by our three SVMs, and then transformed into feature space through the Gram-Schmidt procedure. These results are placed in a 3D histogram, and the function linearly interpolated between bins. A picture of this template is shown (in light blue) together with our original three templates in Figure 9.11. Also shown there are the projection histograms of the these templates onto each SVM's axis.

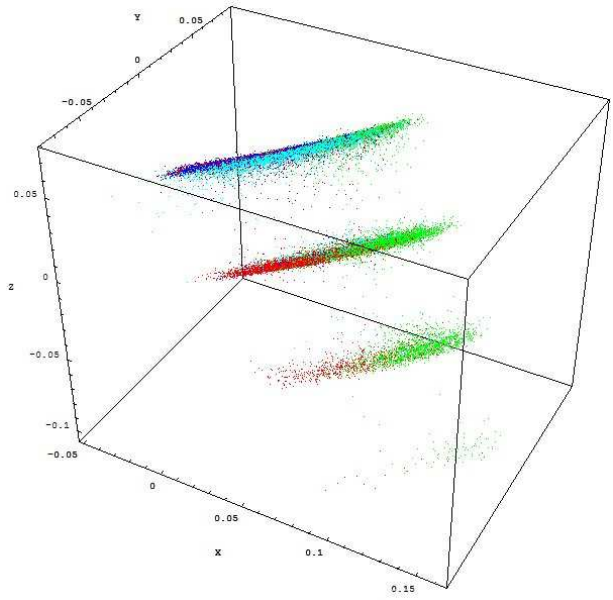


Figure 9.11: Jet Electron Template

	0 tag	1 tag	2 tags	$\geq$ 3 tags
Pythia $t\bar{t}$	0.408183	0.440856	0.144519	0.00644222
Jetele	0.912403	0.0817759	0.00564956	0.000171199

Figure 9.12: b-tag spectrum

When these four templates are used to fit the real data, the maximum likelihood gives a fractional value of zero (0.0) for the jet-electron template. In order to investigate this result, it was theorized that the b-tagging feature of jet-electrons was not very well modeled. To test this idea, the jet-electron pool was recreated twice, assigning b-tags to events randomly such that they would have half as many, and just as many, as the $t\bar{t}$ signal. The $t\bar{t}$ Monte Carlo has the spectrum of tags shown in Figure 9.12.

Once this was done, the likelihood fit to real data still gave a null result when using these two new templates. This seemed to indicate that across all the tag bins for the real data, the events which survive our selection criteria do not conform very well overall to the jet-electron template. To delve into this further, three SVMs were trained to examine how jet-electrons are different from our $t\bar{t}$, W+Jets light flavor, and W+Jets heavy flavor Monte Carlo. Interestingly, it was found that these events are just about perfectly separable from each of the three original classes. Upon taking the gradients of these three new SVMs, the separation comes almost exclusively from a combination of H_T and $\not\!\!E_T$. Perhaps this is to be expected, as these types of events are removed by the initial selection cuts. However, it is difficult to see this *a priori* from the relaxed quantities used to define the jet-electrons.

In order to establish some kind of upper limit on the contribution from this template, the original jet-electron template was used to draw pseudo-experiments which slowly increased the jet-electron fraction of the toy events. It was found that there was a bias of the SVM framework to underestimate the amount of jet-electrons in PEs. It was not until the overall content of the PE reached 12% jet-electrons that our procedure recognized they were present. When this underestimation takes place, note that it is mostly over accounted for in the W+Jets light and heavy flavor classes. From these PEs, it was conservatively estimated that the content of the Real Data represented by this jet-electron template must be less then 12%, and therefore an error in the $t\bar{t}$ fraction θ from this source is something less then 7.5%. To account for QCD fakes, we therefore take an overall systamatic error of -7.5% on $\sigma_{t\bar{t}}$ (*i.e.* a 7.5% systematic on $\sigma_{t\bar{t}}$'s lower error bar).

9.4 Systematic Errors

The following sections will detail the evaluation of the effects of the relaxation of certain assumptions made in the Monte Carlo used to train our Support Vector Machines. Among many models, one must be chosen to actually generate our Monte Carlo and simulate the physics we are interested in studying. The choice of the best model is not always obvious, and it is possible these choices introduce a systematic bias in the measurement we are trying to undertake.

In evaluating the magnitude of these systematic errors, a specific approach was taken in order to be as consistent as possible across all the different sources of error.

It was decided that we ask ourselves the following question: "What value of $\sigma_{t\bar{t}}$ would have been calculated instead, if the potential source of error under investigation were to be adjusted by one standard deviation (1 σ)?"

This means that each quantity that goes into calculating $\sigma_{t\bar{t}}$ in Equation (6.8) is considered in turn. If the quantity has some dependence on the systematic, it will be re-evaluated under conditions that take the shifted systematic into account. These newly calculated quantities are then substituted into Equation (6.8) and the result is compared to our central value for $\sigma_{t\bar{t}}$.

Sticking to this procedure removes any ambiguity as to what is actually being probed when evaluating a systematic. Also, it takes into account correlations between quantities in Equation (6.8) which otherwise may be difficult to account for if the errors on individual terms were considered separately. The final answers for each source of error can be combined in quadrature, and give an overall error estimate on $\sigma_{t\bar{t}}$.

9.4.1 Jet Energy Scale

The reconstruction of hadronic jets in the detector is a complicated process. Not only are there several choices of algorithm for constructing jets, but there are detector effects that must be taken into account in arriving at an overall answer for the energy contained within a jet. All of these corrections are used to arrive at our best guess as to the actual energy content of the hadronic decay at the root of the jet. For a complete discussion of the corrections applied to jets at CDF, see [31],[32], and [33]. The important point to recognize for our discussion here is that there is an error associated with the jet's energy which is determined through the application of these corrections. This error is typically referred to collectively as the Jet Energy Scale (JES) systematic error.

The software simulation of CDF allows one to set the JES at ± 1 σ in the event reconstruction. Applying such a setting causes the jets in an event to get more/less energy than they otherwise might have had given the nominal approach to jet reconstruction. In order to evaluate how the choice of JES affects our cross section measurement, all of our Monte Carlo events were re-evaluated with the jet energies in every event being shifted by $+1\sigma$ and -1σ. Our pseudo-experiment procedure was then repeated, except that the events drawn for each PE were taken from the JES shifted events. In this way, a new batch of linearity PEs was calculated for both JES $+1\sigma$ and JES -1σ. Our templates and everything else about the method remained unchanged from our baseline series of PEs. To estimate the JES systematic error, we can examine

how the biases in the PEs shift with respect to the original baseline pseudo-experiment results. Figure 9.13 shows the results of the pseudo-experiments for JES $+1\sigma$, while Figure 9.14 shows JES -1σ. The Δ on the y-axis shown in these plots has the small bias from our original linearity plots subtracted out. Thus, these plots show the bias from JES only, and not any other effects that may arise from our methodology.

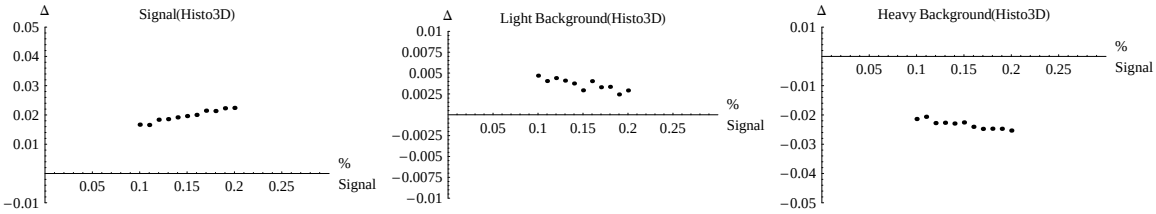


Figure 9.13: JES $+1\sigma$ Pseudo-Experiments

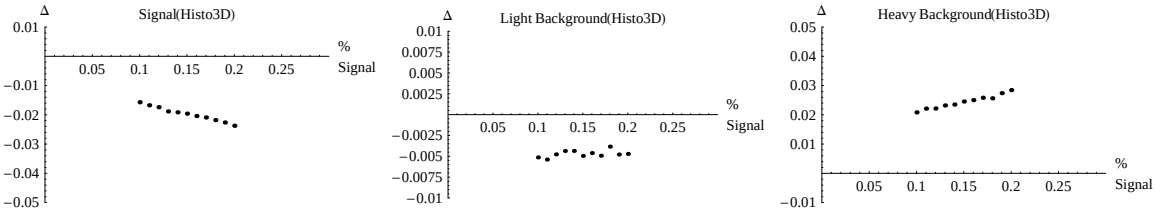


Figure 9.14: JES -1σ Pseudo-Experiments

As can be seen in the Equation (6.8) for the $t\bar{t}$ cross section, other terms of interest are ε for CEM, CMUP, CMX, and their associated luminosities $\mathscr{L}$. Luminosity has no dependence on the jet energy scale. However, the detector acceptance for $t\bar{t}$ does have some reliance on JES. This simply comes from the fact that some of our selection cuts require a missing $E_T > 20$ GeV and ≥ 3 tight jets with energy > 20 GeV. When the jets are JES shifted in energy, there are some number of events which slosh, into or out of, our selection window. See Figure 9.15.

Finally, the number of observed events N_{obs} also has JES dependence. Again, because of our selection criteria, more events come under our scrutiny as we allocate more energy to the jets in our events (and vice versa). While we have reliable Monte Carlo for our $t\bar{t}$ events, N_{obs} will include events from every possible process which can occur in our detector. As such, instead of relying on Monte Carlo to estimate the shift in N_{obs} due to JES, I've taken the real data from our $p\bar{p}$ collisions and applied the JES

shifts to them. As the number of surviving real data events was used as the basis for our PE size of 6,888 events, doing a similar thing here should be a fair way to estimate how the size of N_{obs} changes with JES.

JES has a unique property among our various systematics in that we know which JES we are using in both the Real Data (RD) and Monte Carlo (MC) events. Suppose we were considering the mass of the top quark as a source of systematic error. While we could tweak the mass of the top in our MC to see how our method reacts to each different situation, the Real Data would always contain a top quark with a fixed (though to us uncertain) mass.

Thus, we could not correct for any bias we might find in our method due to top mass because we wouldn't know which mass was the correct one. However, with JES we always know which JES we are using. JES is a correction applied to events, both MC and RD. This allows us to correct for any bias seen in PEs due to the JES we are currently employing.

To estimate the JES uncertainty on $\sigma_{t\bar{t}}$, we calculate $\sigma_{t\bar{t}}$ by plugging into Equation (6.8) the three acceptances $\mathcal{A}_{CEM}$, $\mathcal{A}_{CMUP}$, $\mathcal{A}_{CMX}$ and also N_{obs} at JES = $+1\sigma$. For θ, we can measure it on the RD events reconstructed with JES = $+1\sigma$, and then correct that value by the bias we observe in our apparatus when it encounters JES = $+1\sigma$ events (per Figure 9.13). This gives us a measured value for the cross section at JES = $+1\sigma$. Thus we have arrived at the value for $\sigma_{t\bar{t}}$ that we would have calculated on the real data assuming a differing JES. We repeat this procedure at JES = -1σ. By comparing these two cross sections to our central value, we can estimate our JES systematic uncertainty. See Figure 9.16 for the results of this procedure.

JES Shift	CEM	CMUP	CMX	PHX	BMU
+1 σ	205,314	129,491	55,542	52,208	33,238
0 σ	199,231	125,425	53,785	50,482	32,115
-1 σ	192,702	121,274	51,991	48,628	30,868
	$\Delta\mathcal{A}$ CEM	$\Delta\mathcal{A}$ CMUP	$\Delta\mathcal{A}$ CMX	$\Delta\mathcal{A}$ PHX	$\Delta\mathcal{A}$ BMU
+1 σ	+0.001363	+0.0009114	+0.0003982	+0.0003869	+0.0002517
	+3.05%	+3.24%	+3.27%	+3.42%	+3.50%
-1 σ	-0.001463	-0.0009304	-0.0004065	-0.0004156	-0.0002795
	-3.28%	-3.31%	-3.34%	-3.67%	-3.88%

Total CEM/CMUP Events:	4,461,497
Total CMX Events:	4,412,909

Figure 9.15: $t\bar{t}$ Acceptance at JES ± 1 σ

JES Shift	N	θ_{raw}	Correction	θ	$N_{t\bar{t}}$	$\sigma_{t\bar{t}}$	Δ
$+1\ \sigma$	8,535	0.1570	-0.019	0.1380	1,177.8	7.102 pb	-0.6%
$0\ \sigma$	6,888	0.1667	0.000	0.1667	1,148.2	7.144 pb	-
$-1\ \sigma$	5,729	0.1693	$+0.023$	0.1923	1,101.7	7.085 pb	-0.8%

Figure 9.16: θ & $\sigma_{t\bar{t}}$ calculations at JES $\pm 1\ \sigma$

9.4.2 $t\bar{t}$ Generator

As an alternative to the Pythia event generator used to create the $t\bar{t}$ Monte Carlo, the Herwig generator was used by CDF to create an independent group of some 4×10^6 $t\bar{t}$ events (specifically, dataset otop1s). As was done with the JES systematic, the Herwig events were used to draw pseudo-experiments. In this case Pythia $t\bar{t}$ was replaced by Herwig, and all other aspects of the PE procedure remained constant. The results from this series of pseudo-experiments is shown in Figure 9.18. The acceptance values for Herwig $t\bar{t}$ were also calculated, and are listed in Figure 9.17.

$t\bar{t}$ Gen	CEM	CMUP	CMX	PHX	BMU
Herwig	172,146	108,932	46,834	43,272	27,274
Pythia	199,231	125,425	53,785	50,482	32,115

	$\Delta\mathcal{A}$ CEM	$\Delta\mathcal{A}$ CMUP	$\Delta\mathcal{A}$ CMX	$\Delta\mathcal{A}$ PHX	$\Delta\mathcal{A}$ BMU
	$+0.001607$	$+0.001162$	$+0.0005377$	$+0.0003139$	$+0.0001314$
	$+3.60\%$	$+4.13\%$	$+4.41\%$	$+2.77\%$	$+1.82\%$

Herwig CEM/CMUP Events:	3,721,073	Herwig CMX Events:	3,680,253
Pythia CEM/CMUP Events:	4,461,497	Pythia CMX Events:	4,412,909

Figure 9.17: $t\bar{t}$ Acceptance Herwig vs. Pythia

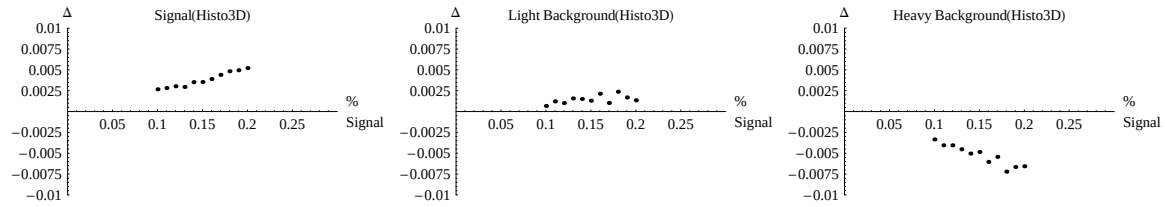


Figure 9.18: Herwig Pseudo-Experiments

9.4.3 Generator PDF

The Parton Distribution Functions used in the creation of our $t\bar{t}$ Monte Carlo can also be varied considerably. As standard procedure, CDF uses the so-called CTEQ5L PDF set. To evaluate this choice's effect on our measurement, a series of 45 other sets were considered as alternatives. The top group at CDF has a standard procedure for re-weighting our nominal Pythia Monte Carlo to account for the differences in each of these PDF sets. See [24] for a full description of this re-weighting procedure. This re-weighting scheme was used to evaluate the $t\bar{t}$ acceptance from each of these sets, and pseudo-experiments were run to see how our measurement of θ is biased. In order to create these PEs, each of our templates was recreated weighting the events differently for each PDF set. The templates were then sampled (treating each as a probability density) in order to create event pools which reflect that template's distribution in feature space. These pools were then randomly sampled when drawing pseudo-experiments. PEs were run at each of the 46 weightings, giving an average θ for each PDF. These shifted θ's, along with the adjusted $\mathcal{A}$'s, were combined to get a value for the cross section for each PDF. CDF has a standard procedure for combining the results from these 46 different PDFs. The resulting errors on ε and θ (and therefore $\sigma_{t\bar{t}}$) are summarized in Figure 9.29.

9.4.4 Q^2

See Figure 9.19 for a summary of the Q^2 systematic error. Two sets of Monte Carlo were used to replace all the W+Jets backgrounds in pseudo-experiments. One modified the Q^2 higher, and the other moved it lower. Two things should be noted here. First, there was no Monte Carlo generated for the tau processes W$\rightarrow \tau\nu$+Jets (lf, bb, cc, and c). These were assumed to not contribute to the classes when working out their fractional sub-process compositions. Second, though usually referred to off-handedly as the $Q^2 = 0.5$ and $Q^2 = 2.0$ shifts, what is actually done is to measure the effect of having $Q^2 \rightarrow (Q/2)^2$ and $Q^2 \rightarrow (2Q)^2$.

9.4.5 *b*-tagging Scale Factor

It is observed that the rate of tagging jets in the real data is less than that found in Monte Carlo estimates. See Figure 9.21. The scale factor which relates the rate in Data to that of Monte Carlo is found to be $\mathrm{SF}_b = 0.95 \pm 0.04$. To correct the Monte

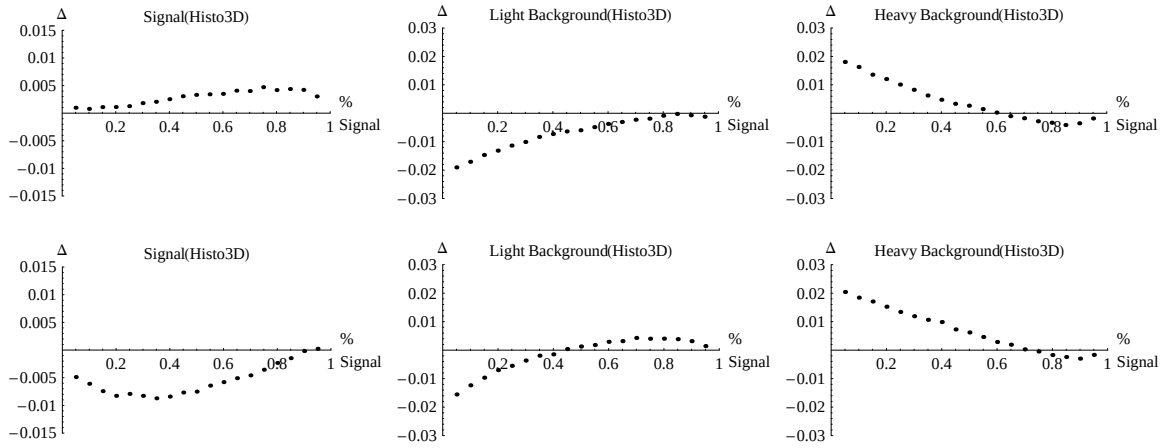


Figure 9.19: $Q^2 = 0.5$ PEs (upper) and $Q^2 = 2.0$ PEs (lower)

	Q^2=0.5	Q^2=2.0
$\Delta\theta$	+0.0017	-0.0070
%	+0.9	-3.9

Figure 9.20: W+Jets Q^2 Systematic

Carlo for this effect, 5% of all heavy (b and c) jets in simulated events are dropped. Specifically, for each b or c jet that was tagged, a random number between 0 and 1 was thrown. If that number happened to be greater then 0.95, this jet was not counted as being tagged in the tally of tags for that event. For light flavor jets (u, d, or s), the so-called "mistag matrix" was used to compute the probability that jet would be (wrongly) tagged. Another random number between 0 and 1 was then selected, and if it was discovered to be less then the probability for mistag, then the tag count for the corresponding event was incremented. This method was followed as standard procedure for all Monte Carlo used in this measurement.

To evaluate a systematic error based on the error bar on the b-tagging scale factor, separate sets of Monte Carlo were generated taking SF_b to be 0.99 and 0.91. These samples were then used to draw pseudo-experiments. Since SF_b doesn't change the event selection, all values of ε have no error associated with this effect. Luminosity is also unaffected. See the systematic error summary in Figure 9.29.

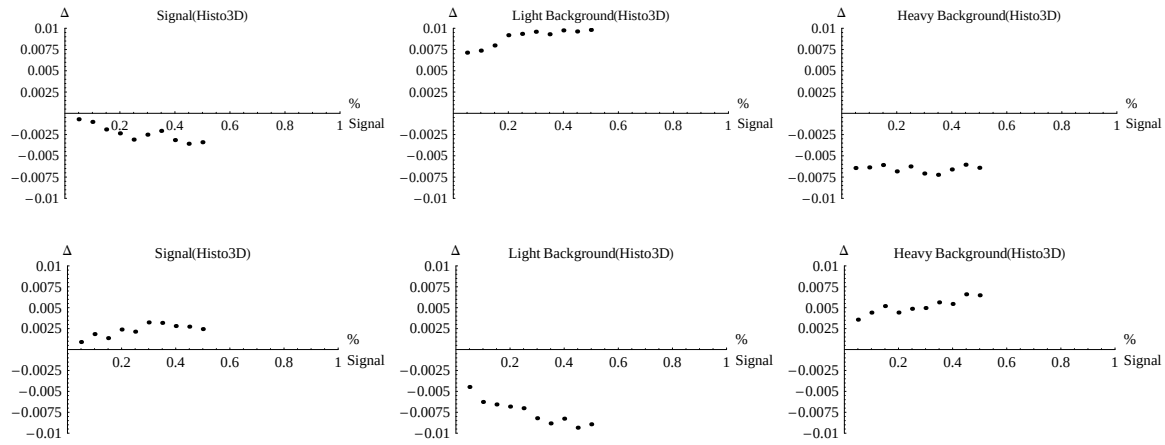


Figure 9.21: $SF_b = 0.91$ PEs (upper) and $SF_b = 0.99$ PEs (lower)

	SF = 0.91	SF = 0.99
$\Delta\theta$	-0.0030	+0.0027
%	-1.7	+1.5

Figure 9.22: b-tag SF Systematic

9.4.6 Initial / Final State Radiation

Processes that contain colored and/or electrically charged objects in their initial or final configurations can emit photon/gluon radiation. Adding these types of corrections to our basic event topologies is necessary to match the observed jet multiplicities in hadron colliders. Initial State Radiation (ISR) refers to radiation from any objects before the main hard scattering event, while Final State Radiation (FSR) refers to any emissions added to the collision products. CDF has MC samples which add/subtract both ISR and FSR for the $t\bar{t}$ signal process.

The ISR/FSR more sample (otop03) was used to evaluate an upper "1σ" systematic. The ISR/FSR less sample (otop04) was used to evaluate the lower "1σ" shift of this systematic. Both are succinctly summarized in Figure 9.23.

9.4.7 Top Quark Mass

All of the $t\bar{t}$ Monte Carlo discussed in this text has been taken to have a mass of 175 GeV/c^2. CDF has generated a series of different mass samples, and following the same

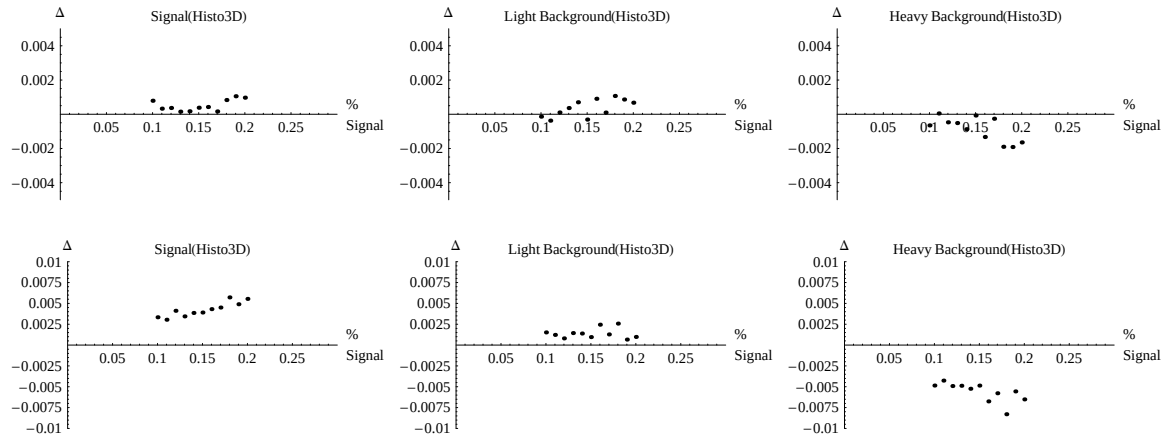


Figure 9.23: ISR/FSR less PEs (upper) and ISR/FSR more PEs (lower)

	ISR&FSR more	ISR&FSR less
ΔCEM	-0.000200119	-0.0001761
%	-0.45	-0.39
ΔCMUP	-0.000225329	-0.000110726
%	-0.80	-0.39
ΔCMX	-0.0000562168	+0.000004900945
%	-0.46	+0.04
$\Delta\theta$	+0.0034	-0.0002
%	+1.9	-0.1

Figure 9.24: ISR/FSR Sys

methodology as above, several different masses were used to investigate how the choice of top mass influences our cross section measurement. At the time of this writing, the best top mass available as measured at the Tevatron is 172.4 ± 1.2 GeV/c^2. PEs were conducted at masses of 165 GeV/c^2, 170 GeV/c^2, and 180 GeV/c^2, such that the events drawn for $t\bar{t}$ and single top took on these values. Top mass is not traditionally cited as a source of error in measurements of the $t\bar{t}$ cross section at CDF. Instead, the dependence of $\sigma_{t\bar{t}}$ on the mass is plotted as a separate entity. See Figure 9.27 for such a plot. The different $t\bar{t}$ acceptance values, along with the θ biases due to changing m_t found from the PEs, are shown in Figure 9.26. Note also that the mass of the top quark was changed in the single top MC used to draw PEs as appropriate.

9.4.8 Luminosity

The overall time integrated luminosity used in this measurement was:

$$\mathscr{L}_{CEM/CMUP} = 2.20 \pm 0.1 \text{ fb}^{-1} \tag{9.9}$$

$$\mathscr{L}_{CMX} = 2.15 \pm 0.1 \text{ fb}^{-1} \tag{9.10}$$

Note that the CMX was inactive in the early part of Run II, which is why it has a slightly lower value for $\mathscr{L}$. Luminosity measured at CDF has an overall 6% error (as shown in the above equations).

9.4.9 Statistical Error and Other Sources

Statistical error in the context of our measurement examines how the value of our measured $t\bar{t}$ cross section depends upon quantities which would simply vary randomly were we to repeat our experiment identically. Because of the finite number of samples taken in our measurement, certain quantities could take on values apart from their true value, simply due to statistical fluctuation in the specific instance of this measurement.

By their very nature, the results of a collider experiment have a random component. When collisions take place, the results and their debris derive from probabilities. By measuring the $t\bar{t}$ cross section, we are trying to determine how often $p\bar{p}$ collisions result in top quarks pairs in lieu of other possibilities. If we were to examine only a handful of collisions, just as if we were to only poll a few potential voters, the answers we calculate would be largely a statistical byproduct of our happenstance. This variance is minimized, of course, by looking at as many collisions as is practical.

All the terms in Equation (6.8) are influenced by this effect. The first term we should consider is N, which is the overall number of events which survive our selection cuts. Just as the number of events from each sub-process of our backgrounds fluctuates as a Poisson, so will the makeup of our overall sample and its signal component. To account for this error, we assign N a statistical error of $\pm\sqrt{N}$. For 6,888 events, this translates to a 1.2% error on N.

The statistical nature of θ was evaluated during the pseudo-experiment process. Each PE represents a simulated execution of our measurement, and so doing many PEs and seeing how θ fluctuates under identical initial conditions will give us a measure of its statistical error. As one might expect, the statistical error on θ is determined

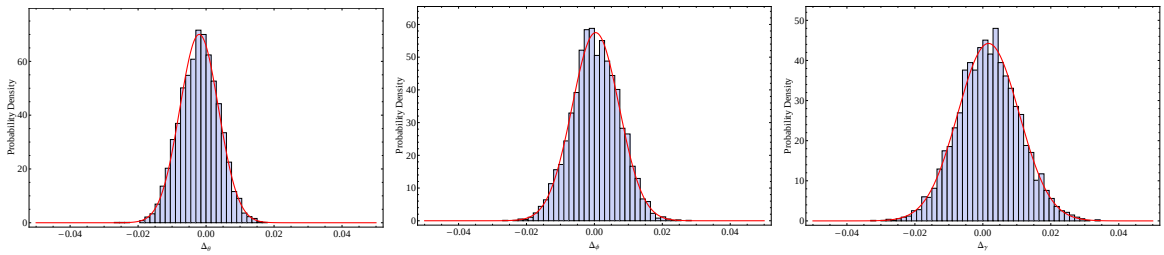


Figure 9.25: (Δ_θ,Δ_ϕ,Δ_γ) from 5k pseudo-experiments

largely by the magnitude of N. However, it isn't any kind of simple dependence, and is tied very closely to our methodology. Since this is true we must use Monte Carlo PEs to calculate its magnitude. The standard deviation of θ as found through these PEs is taken as its statistical error. For our nominal 6,888 events, this standard deviation was found to be 0.0055 for a θ of 0.165 and thereabouts (see linearity plots in Figure 9.8). Figure 9.25 shows a histogram of Δ_θ from a series of 5k PEs, where (θ, ϕ, γ) were fixed to mean values of (0.165, 0.772, 0.063). The histogram in the figure was fit to a Gaussian for comparison.

Figure 9.9 shows the statistical error on θ as a function of the total number of events N in the pseudo-experiment.

9.4.10 Putting It All Together

This section will describe the process of taking all of the different sources of error together. To be clear, when discussing the addition of errors in quadrature for a function $f(x_a, x_b, \ldots)$ with $x_a = a \pm \sigma_a$ and $x_b = b \pm \sigma_b$, *etc.*, it is meant that:

$$\sigma_f = \sqrt{\left(\frac{\partial f}{\partial x_a}\bigg|_{x_a=a}\right)^2 {\sigma_a}^2 + \left(\frac{\partial f}{\partial x_b}\bigg|_{x_b=b}\right)^2 {\sigma_b}^2 + \ldots} \tag{9.11}$$

All the statistical errors on each of the major variables, along with any other quantities which are taken to be completely independent of each other, were added in quadrature. These include the gross errors on the idreco scale factors, trigger efficiencies, and the z-vertex factor ε_{common}.

m_t	CEM	CMUP	CMX	PHX	BMU
165 GeV	25,652	15,969	6,867	6,641	4,242
170 GeV	25,944	16,251	6,944	6,692	4,197
175 GeV	199,231	125,425	53,785	50,482	32,115
180 GeV	26,663	16,727	7,360	6,807	4,322

165 GeV CEM/CMUP Events:	600,829	165 GeV CMX Events:	600,829
170 GeV CEM/CMUP Events:	592,252	170 GeV CMX Events:	580,086
175 GeV CEM/CMUP Events:	4,461,497	175 GeV CMX Events:	4,412,909
180 GeV CEM/CMUP Events:	589,041	180 GeV CMX Events:	589,041

m_t	θ	$\Delta\mathcal{A}$ CEM	$\Delta\mathcal{A}$ CMUP	$\Delta\mathcal{A}$ CMX	$\Delta\mathcal{A}$ PHX	$\Delta\mathcal{A}$ BMU
165 GeV	0.1827	-4.39%	-5.46%	-6.23%	-2.32%	-1.92%
170 GeV	0.1747	-1.90%	-2.40%	-1.78%	-0.14%	-1.55%
175 GeV	0.1667	0.00%	0.00%	0.00%	0.00%	0.00%
180 GeV	0.1603	$+1.36\%$	$+1.01\%$	$+2.52\%$	$+2.13\%$	$+1.93\%$

Figure 9.26: m_t dependence for θ and $t\bar{t}$ Acceptance

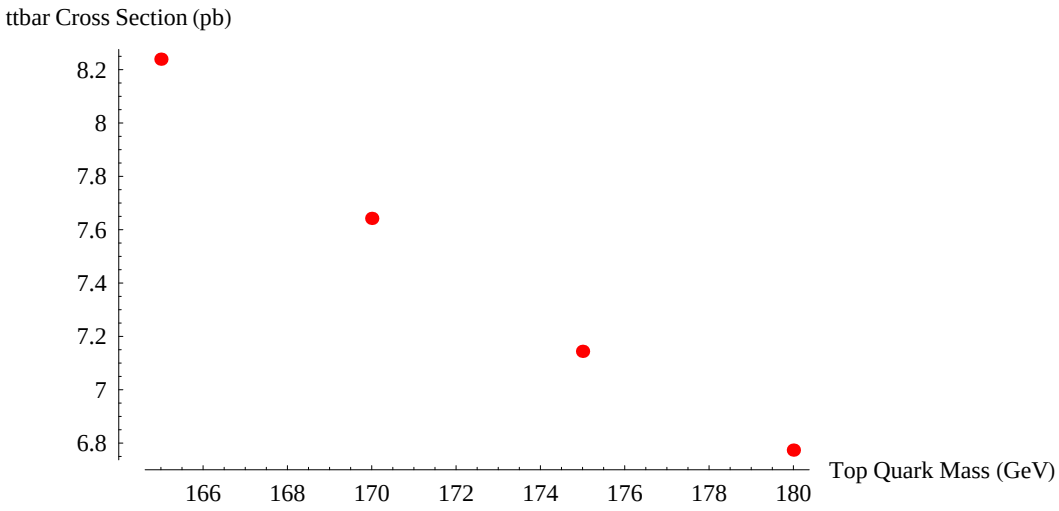


Figure 9.27: $\sigma_{t\bar{t}}$ dependence on top mass

9.5 Application to Real Data

The previous sections have described how an SVM can be used to discriminate $t\bar{t}$ decays from their W+Jets background in Monte Carlo. The general idea is to then take a

trained SVM from Monte Carlo Data, process the real data from the detector with the same initial selection criteria, and then form the maximum likelihood with the events that survive to determine θ. This should then be the fractional amount of signal in the real data sample. We have carried this out using the current release of the real data top ntuples through Run Period 13, which corresponds to an integrated luminosity of 2.2 fb^{-1}.

The results for the kinematic and b-tagging SVMs are summarized in Figure 9.28. Plugging these results into Equation (6.8), we get a central value for our $t\bar{t}$ cross section from the b-tag SMVs of 7.14 pb.

The negative value for γ for the kinematic SVM deserves some explaination. While the actual values for (θ, ϕ, γ) are constrained to be positive and sum to 1, the values shown in the figure are after the baseline correction. The baseline correction for γ is a subtraction, and when combined with the large statistical errors on ϕ and γ for the kinematic SVMs, gives a negative value for γ in the instance of the real data. It shouldn't be taken too seriously, other then the fact that the real data shows a downward fluctuation in γ. The statistical errors are so large that we aren't really interested in the specific information ϕ and γ give in the kinematic case.

SVM	$\theta \pm$ stat	$\phi \pm$ stat	$\gamma \pm$ stat
Kinematic	0.1764 ± 0.0095	0.8476 ± 0.041	-0.0240 ± 0.045
b-tagging	0.1667 ± 0.0055	0.7716 ± 0.0068	0.0617 ± 0.0088

Figure 9.28: (θ, ϕ, γ) for Kinematic and b-tag SVMs

Source	$\mathcal{A}_{CEM}$	$\mathcal{A}_{CMUP}$	$\mathcal{A}_{CMX}$	$\theta_{t\bar{t}}$	N	$\mathscr{L}$	$\mathscr{L}_{CMX}$	Total
JES -1σ	-3.3%	-3.3%	-3.3%	15.3%	-16.8%	-	-	-0.8%
JES $+1\sigma$	$+3.1\%$	$+3.2\%$	$+3.3\%$	-19.2%	$+23.9\%$	-	-	-0.6%
Herwig	$+3.6\%$	$+4.1\%$	$+4.4\%$	-2.5%	-	-	-	-6.1%
ISR/FSR -1σ	-0.4%	-0.4%	0.04%	-0.6%	-	-	-	-0.3%
ISR/FSR $+1\sigma$	-0.4%	-0.8%	-0.5%	-2.4%	-	-	-	-1.9%
PDF -1σ	-0.7%	-0.9%	-0.2%	-1.2%	-	-	-	-1.0%
PDF $+1\sigma$	$+0.4\%$	$+0.6\%$	$+0.2\%$	$+1.0\%$	-	-	-	$+1.3\%$
$\mathrm{SF}_b = 0.91$	-	-	-	$+1.0\%$	-	-	-	$+1.0\%$
$\mathrm{SF}_b = 0.99$	-	-	-	-1.2%	-	-	-	-1.2%
Q^2 less	-	-	-	-0.8%	-	-	-	-0.8%
Q^2 more	-	-	-	$+4.2\%$	-	-	-	$+4.2\%$
Luminosity -1σ	-	-	-	-	-	-6.0%	-6.0%	$+6.4\%$
Luminosity $+1\sigma$	-	-	-	-	-	$+6.0\%$	$+6.0\%$	-5.7%
QCD Fakes	-	-	-	-	-	-	-	-7.5%
Statistical	-	-	-	$\pm 3.3\%$	$\pm 1.2\%$	-	-	$\pm 3.5\%$
Common								$\pm 3.6\%$
All								$\mathbf{+9.3\%}$ $\mathbf{-12.6\%}$

Figure 9.29: Error Summary

Chapter 10

Results

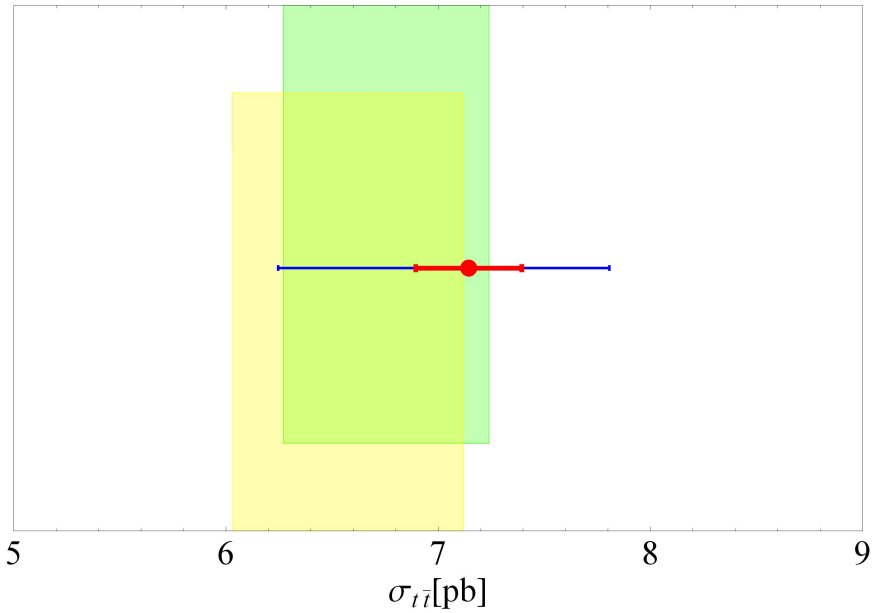


Figure 10.1: Measured $\sigma_{t\bar{t}}$ (stat) *vs.* Theory (Cacciari[26] and Kidonakis[28])

10.1 Central Value

Plugging in the values for acceptance (Figure 6.3), trigger efficiency (Figures 6.8, 6.9, 6.10), idreco scale factors (Figures 6.5, 6.6, 6.7), luminosity (Figure 6.4), z-vertex common factor (Figure 6.11), and theta (Figure 9.28) into Equation (6.8), the overall central value for the cross section is found to be

$$\sigma_{t\bar{t}} = 7.14 \pm 0.25(\text{stat})^{+0.41}_{-0.76}(\text{sys})^{+0.45}_{-0.40}(\text{lumi}) \text{ pb} \tag{10.1}$$

which includes all sources of error summarized in Figure 9.29. Folding the luminosity errors into the systematics:

$$\sigma_{t\bar{t}} = 7.14 \pm 0.25(\text{stat})^{+0.61}_{-0.86}(\text{sys}) \text{ pb} \tag{10.2}$$

and overall:

$$\sigma_{t\bar{t}} = 7.14^{+0.66}_{-0.90} \text{ pb} \tag{10.3}$$

The error bars on $\sigma_{t\bar{t}}$ are asymmetric. It can be symmetrized by taking the values in Equation (10.3) as describing a lopsided Gaussian distribution. We can then find a symmetric Gaussian which has the same weighted mean as the asymmetric version, and a width such that the area under the old function around the new mean is one (1) standard deviation.

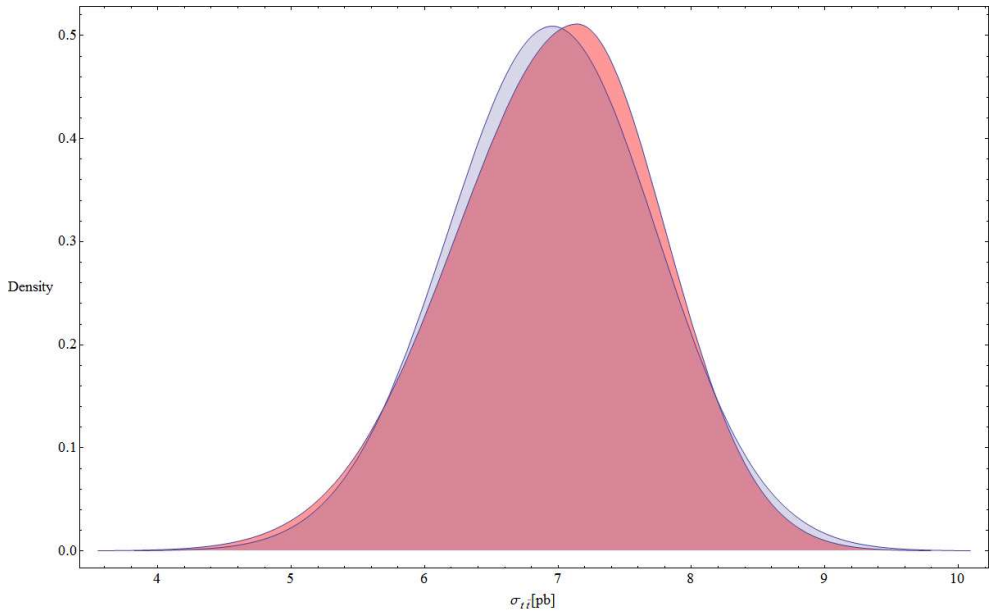


Figure 10.2: Symmetric *vs.* Asymmetric

Specifically, for an asymmetric Gaussian distribution $G(x \mid \mu, \sigma_-, \sigma_+)$, we find the symmetric Gaussian $G(x \mid \mu', \sigma')$, such that:

$$\mu' = \int x\, G(x \mid \mu, \sigma_-, \sigma_+)\, dx \tag{10.4}$$

and $\sigma' = a$ where

$$\int_{\mu'-a}^{\mu'+a} G(x \mid \mu, \sigma_-, \sigma_+)\, dx = 0.6827\ldots \tag{10.5}$$

Following this procedure, it was found that

$$\sigma_{t\bar{t}} = 6.96 \pm 0.78 \text{ pb (symmetrized)} \tag{10.6}$$

Figure 10.2 shows the difference between the symmetric and asymmetric Gaussian distributions. Figure 10.3 breaks down the central value for $\sigma_{t\bar{t}}$ by run period.

Period	Events	CEM	CMUP	CMX	θ	$\sigma_{t\bar{t}}$ [pb]
0d	988	602	267	119	0.1698	6.85
0h	1,133	652	307	174	0.1776	7.44
0i	816	489	193	134	0.1559	6.62
8	536	315	141	80	0.1517	6.63
9	508	306	143	59	0.1702	7.47
10	794	436	232	126	0.1781	7.95
11	749	445	190	114	0.1621	7.23
12	523	313	124	86	0.1525	6.85
13	841	517	201	123	0.1644	6.94
CEM		4,075			0.1571	7.16
CMUP			1,798		0.1957	7.19
CMX				1,015	0.1406	6.36
All	6,888				0.1667	7.14

Figure 10.3: $\sigma_{t\bar{t}}$ by Run Period

10.2 Results Comparison

Figures 10.4 and 10.5 show some of the results from other measurements of $\sigma_{t\bar{t}}$ given by CDF and D0. Figure 10.6 shows some more recent CDF results (which assume a slightly lower value for m_t). From Figure 9.27, it can be seen that the central value of $\sigma_{t\bar{t}}$ for this thesis shifts to 7.4 pb at $m_t = 172.5$ GeV/c^2.

The error due to luminosity is consistent across all measurements, as the 6% error is standard. Note that the newer measurements that quote a smaller error in the luminosity column are actually forming the ratio of $\sigma_{t\bar{t}}$ with $\sigma(Z/\gamma \to \ell\ell)$. The ratio is then multiplied by the theoretical value for $\sigma(Z/\gamma \to \ell\ell)$ to extract $\sigma_{t\bar{t}}$. Thus for those measurements, the last number listed is actually an error due to the theoretical $\sigma(Z/\gamma)$ calculation, and not the delivered luminosity.

Our systematic error is dominated by the QCD fakes and generator (Herwig *vs.* Pythia) errors, and is comparable with other measurements at the 2.2 fb^{-1} level. The QCD fakes error can possibly be reduced through further study. CDF has developed a more advanced QCD veto, and the increase in available data with > 4.0 fb^{-1} should allow our measurement to use a tighter event selection to further reduce this source of error. Both of these options should be explored in any future application of the techniques of this thesis to the full CDF dataset.

The statistical error we quote is one of the best from the Tevatron. This derives mostly from the small statistical error found in computing $\theta_{t\bar{t}}$. Most other measurements in the L+J channel either do not use b-tagging or require a b-tag in their event selection. By keeping our event selection in the pre-tag regime, but utilizing the tag information to extract the signal content (while still differentiating $t\bar{t}$ from W+bb and other heavy backgrounds), we were able to construct an excellent classifier that maintains a large N sample size.

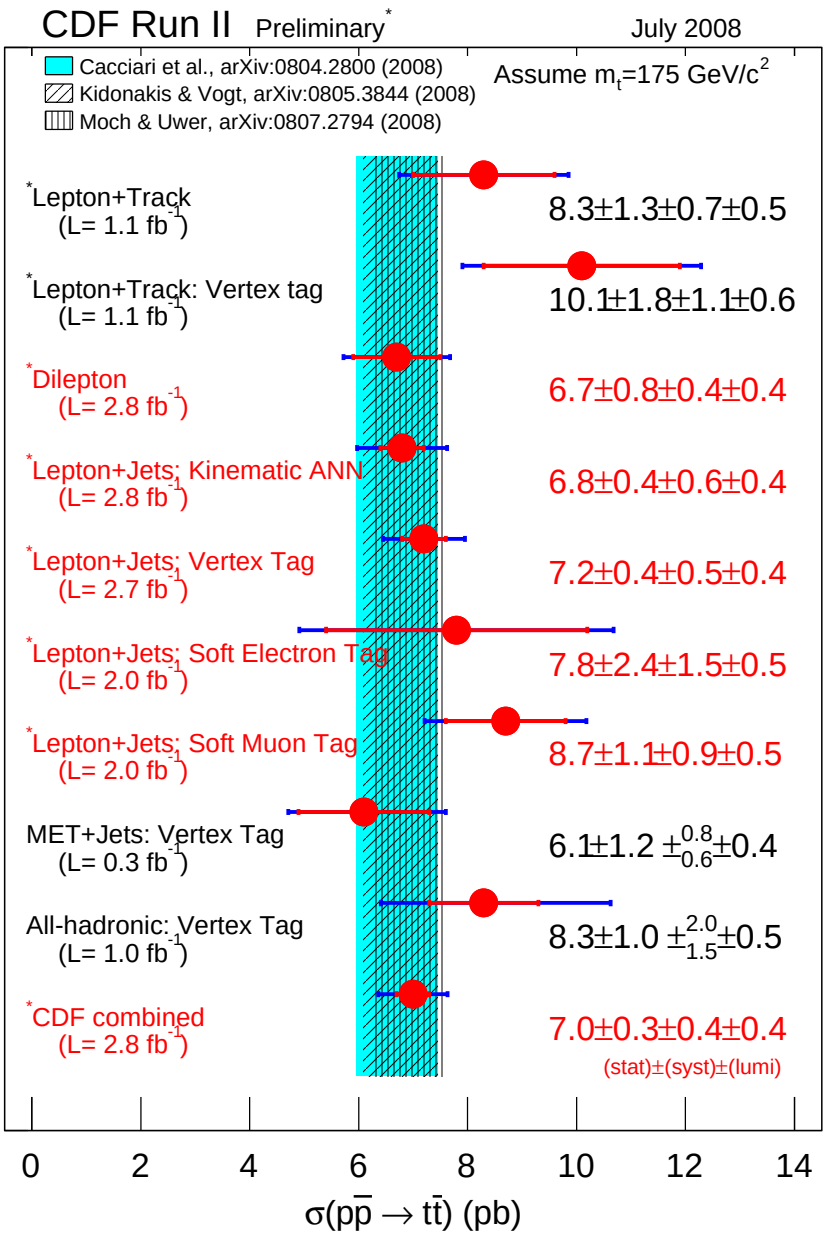


Figure 10.4: $\sigma_{t\bar{t}}$ Measurements at CDF

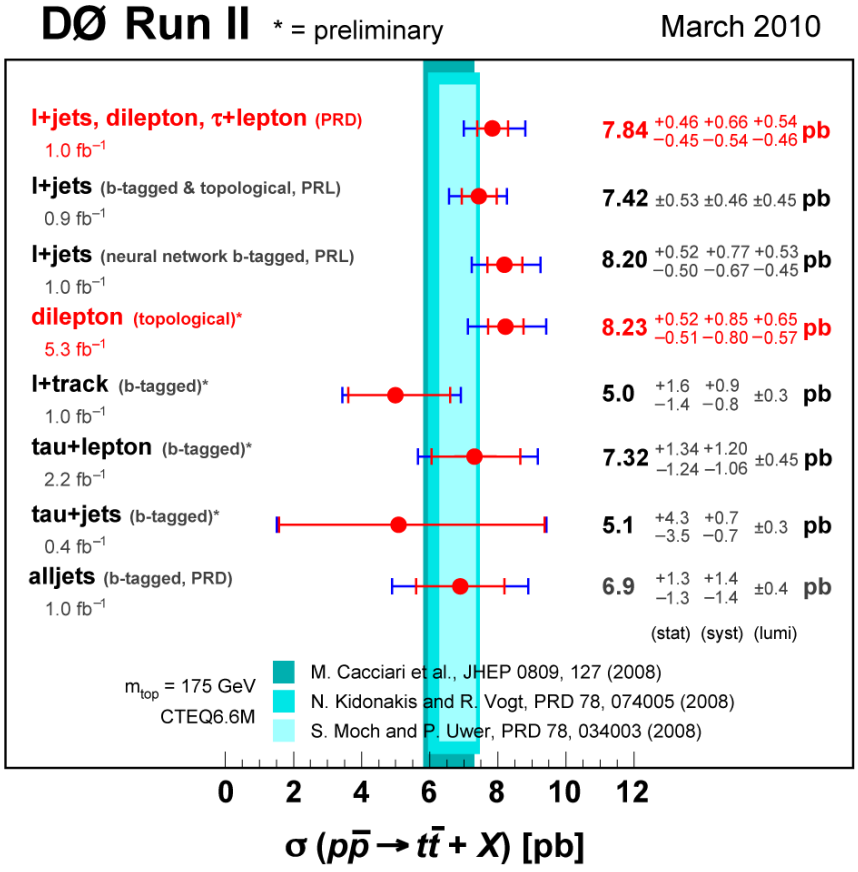


Figure 10.5: $\sigma_{t\bar{t}}$ Measurements at DØ

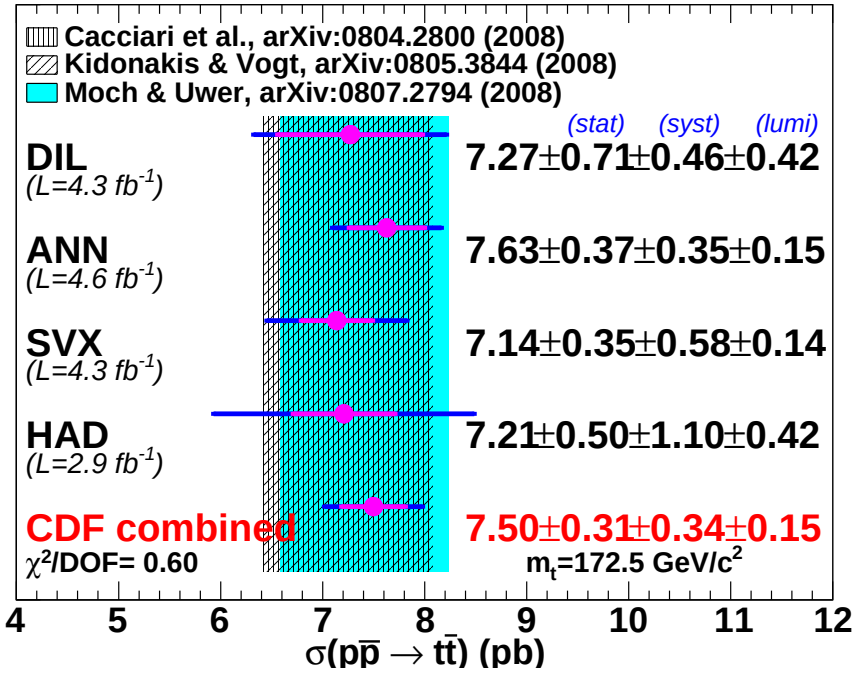


Figure 10.6: $\sigma_{t\bar{t}}$ Measurements at CDF (assuming $m_t = 172.5$ GeV/c^2)

Part V

Appendices

Appendix A

Gram-Schmidt Expansion

This appendix is designed to expound the use of the Gram-Schmidt procedure using SVM decision functions as the basis vectors of the expansion. In feature space, the decision plane of a trained SVM can be expressed as follows[1]:

$$|w\rangle = \sum_{i}^{SV} y_i \alpha_i |\varphi(\vec{x_i})\rangle \tag{A.1}$$

Here, we mean $\vec{x_i}$ to be that SVM's i^{th} support vector, and y_i and α_i to be its associated target value and Lagrange multiplier.

Given a set of three SVMs that use the same mapping function φ to feature space, the Gram-Schmidt prescription results in the following orthonormal basis vectors:

$$|w_1'\rangle = |w_1\rangle \tag{A.2}$$

$$|w_2'\rangle = |w_2\rangle - \frac{\langle w_1|w_2\rangle}{\langle w_1|w_1\rangle}|w_1\rangle \tag{A.3}$$

$$|w_3'\rangle = |w_3\rangle - \frac{\langle w_1|w_3\rangle}{\langle w_1|w_1\rangle}|w_1\rangle - \frac{\langle w_2'|w_3\rangle}{\langle w_2'|w_2'\rangle}|w_2'\rangle \tag{A.4}$$

The three dimensional point {X,Y,Z} in feature space corresponding to a test vector $\vec{z}$ in our original un-mapped space can then be computed as:

$$X(\vec{z}) = \frac{\langle w_1'|\varphi(\vec{z})\rangle}{\langle w_1'|w_1'\rangle} \tag{A.5}$$

$$Y(\vec{z}) = \frac{\langle w_2' | \varphi(\vec{z}) \rangle}{\langle w_2' | w_2' \rangle} \tag{A.6}$$

$$Z(\vec{z}) = \frac{\langle w_3' | \varphi(\vec{z}) \rangle}{\langle w_3' | w_3' \rangle} \tag{A.7}$$

Recall that to carry out an inner product in feature space defined by the mapping φ, we use the kernel function of that space:

$$\langle \varphi(\vec{a}) | \varphi(\vec{b}) \rangle = K(\vec{a}, \vec{b}) \tag{A.8}$$

To make the computations of Equations (A.5) - (A.7) easier, we will enumerate all the inner products involving $\{w_1, w_2, w_3, w_1', w_2', w_3'\}$ not having any dependence on the test point $\vec{z}$. Here we will use the short hand $a \cdot b = \langle a | b \rangle$, and $a^2 = a \cdot a$.

$$w_1{}^2 = w_1'{}^2 = \sum_{i,j} y_{1i} y_{1j} \alpha_{1i} \alpha_{1j} K_{ij} \tag{A.9}$$

$$w_2{}^2 = \sum_{i,j} y_{2i} y_{2j} \alpha_{2i} \alpha_{2j} K_{ij} \tag{A.10}$$

$$w_3{}^2 = \sum_{i,j} y_{3i} y_{3j} \alpha_{3i} \alpha_{3j} K_{ij} \tag{A.11}$$

$$w_1 \cdot w_2 = \sum_{i,j} y_{1i} y_{2j} \alpha_{1i} \alpha_{2j} K_{ij} \tag{A.12}$$

$$w_1 \cdot w_3 = \sum_{i,j} y_{1i} y_{3j} \alpha_{1i} \alpha_{3j} K_{ij} \tag{A.13}$$

$$w_2 \cdot w_3 = \sum_{i,j} y_{2i} y_{3j} \alpha_{2i} \alpha_{3j} K_{ij} \tag{A.14}$$

Here we mean that $K_{ij} = K(\vec{a_i}, \vec{b_j})$, where $\vec{a_i}$ and $\vec{b_j}$ are the i^{th} and j^{th} support vectors of the SVM(s) involved in the sum. Then,

$$w_2'{}^2 = w_2{}^2 - \frac{(w_1 \cdot w_2)^2}{w_1{}^2} \tag{A.15}$$

$$w_2' \cdot w_3 = w_2 \cdot w_3 - \frac{(w_1 \cdot w_2)(w_1 \cdot w_3)}{w_1{}^2} \tag{A.16}$$

$$w_3'^2 = w_3 \cdot w_3 - \frac{(w_1 \cdot w_3)^2}{w_1^2} - \frac{(w_2' \cdot w_3)^2}{w_2'^2} \tag{A.17}$$

and by construction,

$$w_1 \cdot w_2' = 0 \tag{A.18}$$

$$w_1 \cdot w_3' = 0 \tag{A.19}$$

$$w_2' \cdot w_3' = 0 \tag{A.20}$$

Finally, for the general point $\vec{z}$ we have

$$w_1 \cdot \varphi(\vec{z}) = \sum_i y_{1i}\alpha_{1i}K(\overrightarrow{x_{1i}}, \vec{z}) \tag{A.21}$$

$$w_2' \cdot \varphi(\vec{z}) = \sum_i y_{2i}\alpha_{2i}K(\overrightarrow{x_{2i}}, \vec{z}) - \frac{(w_1 \cdot w_2)(w_1 \cdot \varphi(\vec{z}))}{w_1^2} \tag{A.22}$$

$$w_3' \cdot \varphi(\vec{z}) = \sum_i y_{3i}\alpha_{3i}K(\overrightarrow{x_{3i}}, \vec{z}) - \frac{(w_1 \cdot w_3)(w_1 \cdot \varphi(\vec{z}))}{w_1^2} - \frac{(w_2' \cdot w_3)(w_2' \cdot \varphi(\vec{z}))}{w_2'^2} \tag{A.23}$$

Note the recursive nature of these equations, as the result of Equation (A.21) appears in the second term of (A.22) and (A.23), and Equation (A.22) in (A.23)'s third term. From these equations, it is straight forward to calculate the {X,Y,Z} of Equations (A.5),(A.6), and (A.7).

Appendix B

Poisson Distributed Random Variables

This appendix gives a proof that a set formed by taking samples from two independent Poisson distributed variables is also Poisson distributed. Here, we will use the notation that follows something like $P(X = x) = \frac{1}{10}$, which in words can be read as "The probability that the random variable X takes on the value x is ten percent." We use X to represent a random variable, and it stands in the abstract for any of its possible outcomes. We then use x to indicate the concrete realization of one such random measurement of X.

Given two Poisson distributed variables X_1 and X_2, with corresponding mean parameters λ_1 and λ_2, what is the probability density function of $Y = X_1 + X_2$?

$$P(Y = n) = P(X_1 + X_2 = n) \tag{B.1}$$

$$= \sum_{k=0}^{n} P(X_1 = k)P(X_2 = n - k) \tag{B.2}$$

$$= \sum_{k=0}^{n} e^{-\lambda_1} \frac{\lambda_1^k}{k!} e^{-\lambda_2} \frac{\lambda_2^{n-k}}{(n-k)!} \tag{B.3}$$

$$= e^{-(\lambda_1+\lambda_2)} \sum_{k=0}^{n} \frac{1}{k!(n-k)!} \lambda_1^k \lambda_2^{n-k} \tag{B.4}$$

$$= \frac{1}{n!} e^{-(\lambda_1+\lambda_2)} \sum_{k=0}^{n} \frac{n!}{k!(n-k)!} \lambda_1^k \lambda_2^{n-k} \tag{B.5}$$

$$= e^{-(\lambda_1+\lambda_2)} \frac{(\lambda_1+\lambda_2)^n}{n!} \tag{B.6}$$

We recall in the last step the binomial theorem, which reads:

$$(x+y)^n = \sum_{k=0}^{n} \frac{n!}{k!(n-k)!} x^{n-k} y^k \tag{B.7}$$

Hence, the sum of two Poisson distributed random variables is again Poisson distributed with parameter equal to the sum of the two constituent distributions. This result recursively generalizes such that the sum of N independent random Poisson X's is Poisson with parameter given by $\sum_{i=1}^{N} \lambda_i$.

Appendix C

Light & Heavy Class Composition

The following tables show the individual processes that were used in constructing the Light and Heavy background classes. The processes are grouped together by their underlying physical origin. The column "Fraction" lists the fractional composition for each sub-process within that process block. The rightmost column "All" shows the overall fractional composition for that block, as compared to the other process blocks within the class template. For the Light class, the three blocks were W+Jets, Z+Jets, and dibosons, while in the Heavy class the blocks correspond to W+Jets, Z+Jets, and single top.

Process	Dataset	x-sec(pb)	Fraction	All
W→ eν+bb+0p	btop0w	2.980	0.029617024	0.449102508
W→ eν+bb+1p	btop1w	0.888	0.152637014	
W→ eν+bb+2p	btop2w	0.287	0.321669261	
W→ $\mu\nu$+bb+0p	btop5w	2.980	0.025218874	
W→ $\mu\nu$+bb+1p	btop6w	0.889	0.133827962	
W→ $\mu\nu$+bb+2p	btop7w	0.286	0.278660237	
W→ $\tau\nu$+bb+0p	dtop0w	2.980	0.003341731	
W→ $\tau\nu$+bb+1p	dtop1w	0.888	0.017690943	
W→ $\tau\nu$+bb+2p	dtop2w	0.287	0.037336955	
Z→ ee+bb+0p	ztopb0	0.551	0.069326301	0.028294533
Z→ ee+bb+1p	ztopb1	0.134	0.106773854	
Z→ ee+bb+2p	ztopb2	0.551	0.111770917	
Z→ $\mu\mu$+bb+0p	ztopb5	0.0385	0.041923083	
Z→ $\mu\mu$+bb+1p	ztopb6	0.134	0.137162971	
Z→ $\mu\mu$+bb+2p	ztopb7	0.0385	0.249475049	
Z→ $\tau\tau$+bb+0p	ztopbt	0.625	0.283567824	
single top, s-channel		0.88	0.434991294	0.522602959
single top, t-channel		1.98	0.565008706	

Figure C.1: Heavy class summary

Process	Dataset	x-sec(pb)	Fraction	All
W→ eν+0p	ptopw0	1800	0.004684158	0.870654143
W→ eν+1p	ptopw1	225.0	0.010387354	
W→ eν+2p	ptop2w	35.30	0.060159273	
W→ eν+3p	ptop3w	5.59	0.262711467	
W→ eν+4p	ptoprw	1.03	0.096195014	
W→ $\mu\nu$+0p	ptopw5	1800	0.003891884	
W→ $\mu\nu$+1p	ptopw6	225.0	0.010431707	
W→ $\mu\nu$+2p	ptop7w	35.30	0.052784323	
W→ $\mu\nu$+3p	ptop8w	5.59	0.230790812	
W→ $\mu\nu$+4p	ptop9w	1.03	0.085403204	
W→ $\tau\nu$+0p	utopw0	1800	0.000500299	
W→ $\tau\nu$+1p	utopw1	225.0	0.001038589	
W→ $\tau\nu$+2p	utop2w	35.40	0.005612156	
W→ $\tau\nu$+3p	utop3w	5.60	0.027985075	
W→ $\tau\nu$+4p	utop4w	1.03	0.01112805	
W→ eν+cc+0p	ctop0w	5.00	0.001706753	
W→ eν+cc+1p	ctop1w	1.79	0.009333996	
W→ eν+cc+2p	ctop2w	0.628	0.026771642	
W→ $\mu\nu$+cc+0p	ctop5w	5.00	0.001529929	
W→ $\mu\nu$+cc+1p	ctop6w	1.79	0.00825559	
W→ $\mu\nu$+cc+2p	ctop7w	0.628	0.023455275	
W→ $\tau\nu$+cc+0p	etopw0	5.00	0.000222678	
W→ $\tau\nu$+cc+1p	etopw1	1.80	0.001051447	
W→ $\tau\nu$+cc+2p	etopw2	0.628	0.0030596	
W→ eν+c+0p	stopw0	17.10	0.001128891	
W→ eν+c+1p	stopw1	3.39	0.004919126	
W→ eν+c+2p	stopw2	0.507	0.017589217	
W→ eν+c+3p	stopw3	0.083	0.007259374	
W→ $\mu\nu$+c+0p	stopw5	17.10	0.000802717	
W→ $\mu\nu$+c+1p	stopw6	3.39	0.004236985	
W→ $\mu\nu$+c+2p	stopw7	0.507	0.015190844	
W→ $\mu\nu$+c+3p	stopw8	0.083	0.006296487	
W→ $\tau\nu$+c+0p	stopwa	17.10	0.000112235	
W→ $\tau\nu$+c+1p	stopwb	3.39	0.00054588	
W→ $\tau\nu$+c+2p	stopwc	0.507	0.001965036	
W→ $\tau\nu$+c+3p	stopwd	0.083	0.000862931	

Figure C.2: Light class summary, part 1

Process	Dataset	x-sec(pb)	Fraction	All
Z→ ee+0p	ztopp0	158	0.005774698	0.054531259
Z→ ee+1p	ztopp1	21.6	0.022353636	
Z→ ee+2p	ztop2p	3.47	0.098849848	
Z→ ee+3p	ztop3p	0.55	0.048744146	
Z→ ee+4p	ztop4p	0.0992	0.017133729	
Z→ $\mu\mu$+0p	ztopp5	158	0.007645735	
Z→ $\mu\mu$+1p	ztopp6	21.6	0.014151397	
Z→ $\mu\mu$+2p	ztop7p	3.47	0.0740248	
Z→ $\mu\mu$+3p	ztop8p	0.55	0.166775594	
Z→ $\mu\mu$+4p	ztop9p	0.0992	0.060281419	
Z→ $\tau\tau$+0p	ztopt3	158	0.002317325	
Z→ $\tau\tau$+1p	ztopt4	21.5	0.025620647	
Z→ $\tau\tau$+2p	ztopt2	4.14	0.390732861	
Z→ ee+cc+0p	ctop0w	1.08	0.003328643	
Z→ ee+cc+1p	ctop1w	0.331	0.005558777	
Z→ ee+cc+2p	ctop2w	0.107	0.008694607	
Z→ $\mu\mu$+cc+0p	ctop5w	1.08	0.002300994	
Z→ $\mu\mu$+cc+1p	ctop6w	0.332	0.008624698	
Z→ $\mu\mu$+cc+2p	ctop7w	0.107	0.021276104	
Z→ $\tau\tau$+cc+0p	etopw0	1.28	0.015810344	
diboson WW	itopww	12.40	0.835733126	0.074814598
diboson WZ	itopwz	3.70	0.141737725	
diboson ZZ	itopzz	3.80	0.022529149	

Figure C.3: Light class summary, part 2

Appendix D

MINUIT Transform

D.1 Introduction

The minimization program known as MINUIT is very popular among physicists. One drawback of this program is that it is unable to minimize with respect to parameters that have general mutual constraints. A particular constraint that often appears is the desire for the parameters of a likelihood function to sum to unity (1.0). Explicitly, with n parameters $x_1 \ldots x_n$, we might desire that:

$$\sum_{i=1}^{n} x_i = 1 \tag{D.1}$$

where individually, $0 \leq x_i \leq 1$.

This appendix describes a variable transformation that can be applied such that MINUIT is able to handle these situations.

D.2 Transformation

Because of the constraint given in Equation (D.1), our transformation takes us from a likelihood function of n parameters to another containing only $(n-1)$ parameters. The convention we will use is that the x's will be the parameters of the untransformed likelihood function, and a's will represent the $(n-1)$ transformed parameters.

The transformation is given by

$$\begin{aligned} x_1 &\to a_1 \\ x_2 &\to (1-a_1)a_2 \\ x_3 &\to (1-a_1)(1-a_2)a_3 \\ &\dots \\ x_n &\to (1-a_1)(1-a_2)_{\dots}(1-a_{n-1}) \end{aligned} \tag{D.2}$$

with each of the a's taken to fall on the interval $0 \leq a_i \leq 1$. If we make the definition that $a_n \equiv 1$, then a general transformation can be written as

$$x_i \to a_i \prod_{j=1}^{i-1} (1-a_j) \tag{D.3}$$

and from this, a recursion relation can be written as

$$\frac{x_{i+1}}{x_i} = \frac{a_{i+1}(1-a_i)}{a_i} \tag{D.4}$$

For reference, we note that the inverse transformation is given by

$$\begin{aligned} a_1 &\to x_1 = 1 - \sum_{j=2}^{n} x_j \\ &\dots \\ a_i &\to \frac{x_i}{x_i + \ldots + x_n} \\ &\dots \\ a_{n-1} &\to \frac{x_{n-1}}{x_{n-1} + x_n} \end{aligned} \tag{D.5}$$

Considering the constraint of Equation (D.1), we can see that the middle relationship shown in Equation (D.5) is general in its form.

D.3 Gradient

The gradient of this transformation can be explicitly calculated in terms of the original function's partial derivatives.

The full derivative of our function f with respect to one of our transformed variables a_i is given below. The full gradient is given by computing each of these expressions for the $(n-1)$ variables a.

$$\frac{df}{da_i} = \sum_{j=1}^{n} \frac{\partial f}{\partial x_j} \frac{\partial x_j}{\partial a_i} \tag{D.6}$$

Using Equation (D.2), we can compute the terms $\frac{\partial x_j}{\partial a_i}$. These partial derivatives make up an n x $(n-1)$ matrix. To observe how this matrix behaves for an arbitrary value of n, the $n = 3$, $n = 4$, and $n = 5$ cases are written explicitly below:

$$\begin{matrix} 1 & 0 \\ -a_2 & (1-a_1) \\ -(1-a_2) & -(1-a_1) \end{matrix}$$

$$\begin{matrix} 1 & 0 & 0 \\ -a_2 & (1-a_1) & 0 \\ -(1-a_2)a_3 & -(1-a_1)a_3 & (1-a_1)(1-a_2) \\ -(1-a_2)(1-a_3) & -(1-a_1)(1-a_3) & -(1-a_1)(1-a_2) \end{matrix}$$

$$\begin{matrix} 1 & 0 & 0 & 0 \\ -a_2 & (1-a_1) & 0 & 0 \\ -(1-a_2)a_3 & -(1-a_1)a_3 & (1-a_1)(1-a_2) & 0 \\ -(1-a_2)(1-a_3)a_4 & -(1-a_1)(1-a_3)a_4 & -(1-a_1)(1-a_2)a_4 & (1-a_1)(1-a_2)(1-a_3) \\ -(1-a_2)(1-a_3)(1-a_4) & -(1-a_1)(1-a_3)(1-a_4) & -(1-a_1)(1-a_2)(1-a_4) & -(1-a_1)(1-a_2)(1-a_3) \end{matrix}$$

First, notice that these matrices fill only the lower left triangle. Next, notice that the first $n-1$ rows of the matrix are different then the last. This obviously derives from the asymmetry of the last equation of our transformation. The first $n-1$ rows follow a pattern such that it can be formed by taking the expression for the j^{th} row as: $(-a_j)\prod_{i=1}^{j-1}(1-a_i)$, and then in the k^{th} column remove the associated $(-a_k)$ or $(1-a_k)$ term from this expression. For the n^{th} row, the expression is $-\prod_{i=1}^{n-1}(1-a_i)$, where the $(1-a_k)$ term is removed in the k^{th} column. This Jacobian matrix can then be used in computing the full gradient in Equation (D.6).

The discussion that follows will make the following assumption for the form of f, which is the one I've been employing in a recent experiment. Suppose f^* has the following form:

$$f^*(\vec{x_i}|\,\vec{\theta}\,) = \prod_{i=1}^{N} (\sum_{j=1}^{p} (\vec{\theta}\,)_j (\vec{x}\,_i)_j) \tag{D.7}$$

Here, $\vec{x}$ and $\vec{\theta}$ are vectors of length p. The function f is related to f^* as $f = -\ln(f^*)$, such that:

$$f(\vec{x_i}|\vec{\theta}) = -\sum_{i=1}^{N} \ln(\sum_{j=1}^{p} (\vec{\theta})_j (\vec{x}_i)_j) \tag{D.8}$$

Suppose we are using MINUIT to maximize f with respect to the θ_i's. Each of the N vectors $\vec{x}$ represent a data point where each dimension of x describes the probability density for that data point to derive from the i^{th} class. We would like to impose that $\sum \theta_i = 1$, as then the values that the θ's take at the maximum of f are the most likely fractional composition values for the different classes given these N data points.

Using the transformation we have described in the beginning of this document to impose our constraint on $\vec{\theta}$, the gradient is easily calculated for MINUIT using Equation (D.6) if we calculate $\frac{\partial f}{\partial \vec{\theta}}$. For our function f, we have:

$$\frac{\partial f}{\partial \theta_z} = -\sum_{i=1}^{N} \frac{(\vec{x}_i)_z}{(\sum_{j=1}^{p} (\vec{\theta})_j (\vec{x}_i)_j)} \tag{D.9}$$

D.4 Implementations

This section gives a C++ implementation of our coordinate transformation and the calculation of its gradient.

Transformation

The following computes our transformation:

```
vector<double>
MaxLikeFCN::transformPar( const vector<double>& par ) const
{

  // This transformation lets you impose a constraint such that
  // 0 <= x[i] <= 1 and Sum( x[i] ) = 1
  //
  // Take N-1 variables 0 <= a[i] <= 1, and define a[N] = 1
  //
  // Therefore:
  //
  // x[1] = a[1]
```

```
// x[2] = (1 - a[1]) a[2]
// x[3] = (1 - a[1]) (1 - a[2]) a[3]
// ...
// x[N] = (1 - a[1]) (1 - a[2]) ... (1 - a[N-1]) a[N]
//
// Also note that:
//
//      x[i]/x[i-1] = ( 1 - a[i-1] ) a[i] / a[i-1]
//
// which makes for a quick recursive definition if a[i-1] isn't zero!
//


// take the input vector, and add a[N] = 1 to
// make it the same length as x

vector<double> a( par );
a.push_back( 1.0 );

vector<double> x( a.size() );


// use the recursive definition, unless
// we get into divide by zero trouble...
x[0] = a[0];

for( unsigned i = 1; i < x.size(); ++i )
{
  if( a[i-1] != 0. )
  {
    x[i] = x[i-1] * (1. - a[i-1]) * a[i] / a[i-1];
  }
  else
  {
    x[i] = a[i];
```

```
      for( unsigned j = 0; j < i; ++j )
      {
        x[i] *= (1. - a[j]);
      }
    }
  }

  return x;
}
```

Gradient

The following computes the gradient of the function f described in Equation (D.8). The first half of this code which computes the Jacobian for the general transformation in this document is general, and can be applied toward any function f.

```
vector<double>
MaxLikeFCNGrad::Gradient( const vector<double> & x ) const
{
    //we need the gradient of the Minuit transformed parameters
    //
    //  df/dx[i] = Sum_j( df/dxPrime[j] dxPrime[j]/dx[i] )

    // these are the non-transformed coordinates
    vector<double> xPrime = transformPar( x ) ;

    // this will be the returned gradient
    // ie g[i] = df/dx[i]
    vector<double> g( x.size(), 0. );

    // this is the partial derivative matrix for our transformation
    // partial[i][j] = dxPrime[i]/dx[j]
    vector< vector<double> > partial;

    // compute the partial derivatives of the untransformed
    // coordinates wrt the transformed ones
```

```
for( unsigned i = 0; i < xPrime.size(); ++i )
{
  vector<double> temp( x.size(), 0. );
  if( i < x.size() )
  {
    for( unsigned j = 0; j <= i; ++j )
    {
      temp[j] = 1.;
      for( unsigned k = 0; k <= i; ++k )
      {
        if( k != j )
        {
          if( k == i )
          {
            temp[j] *= -1. * x[k];
          }
          else
          {
            temp[j] *= 1. - x[k];
          }
        }
      }
    }
  }
  else
  {
    // last row of derivatives
    for( unsigned j = 0; j < x.size(); ++j )
    {
      temp[j] = -1.;
      for( unsigned k = 0; k < x.size(); ++k )
      {
        if( k != j )
        {
          temp[j] *= 1. - x[k];
```

```
        }
      }

    } // for
  } // else

  partial.push_back( temp );
}

// Un-comment to print the partials matrix
/*
for(unsigned i = 0; i< partial.size();++i)
{
  for(unsigned j = 0; j < partial[i].size(); ++j )
  {
    cout << partial[i][j] << " ";
  }
  cout << endl;
}
*/


// now compute the gradients
//     each gradient has the same denominator
//     so we compute that first and save it
//     then, each direction has its own numerator
//     we compute all of those, and then update
//     the gradients by adding numerator / denom
for( unsigned i = 0; i < _prob.size(); ++i )
{

  // get the denominator term first
  double denom = 0.;
  for( unsigned j = 0; j < xPrime.size(); ++j )
  {
```

```
        denom += xPrime[j] * _prob[i][j];
      }

      vector<double> numerator( x.size(), 0. );
      for( unsigned j = 0; j < numerator.size(); ++j )
      {
        for( unsigned k = j; k < xPrime.size(); ++k )
        {
          numerator[j] += _prob[i][k] * partial[k][j];
        }
      }

      for(unsigned k = 0; k < numerator.size(); ++k )
      {
        g[k] -= numerator[k] / denom;
      }

    }

    return g;
}
```

Bibliography

[1] Borgman, Jacob. *Visualizing Trichotomous Classification Problems.* Private notes and correspondence.

[2] Cottingham, W.N. & Greenwood, D.A. *An Introduction to the Standard Model of Particle Physics.* Cambridge University Press, Cambridge, UK (1998).

[3] Hastie, T. & Tibshirani, R. *Classification by Pairwise Coupling.* Advances in Neural Information Processing Systems 10, M. I. Jordan, M. J. Kearns, S. A. Solla, eds., MIT Press (1998)

[4] Bloom, K. *Study of the heavy flavor content of jets produced in association with W bosons in $p\bar{p}$ collisions at $\sqrt{s} = 1.8$ TeV.* Phys Rev D. 65 052007 (2002)

[5] Liss, Tony M. & Tipton, Paul L. (Sept. 1997) *The Discovery of the Top Quark.* Scientific American v. 277 p. 54-9

[6] Konsigsberg, J. et al. *Luminosity Information in Run II.* CDF Internal Note 4831 (1999)

[7] Acosta, D. et al. *A First Look at the CLC Luminosity Measurements.* CDF Note 6052 (2002)

[8] Borgman, Jacob. Private Correspondence & Discussions

[9] Perricone, Mike. (February 2006) *Symmetry.* **http://www.symmetrymagazine.org/cms/?pid=1000258**

[10] Fermilab Accelerators: Proton Source. **http://www-bd.fnal.gov/public/proton.html**

[11] Fermilab Accelerators: Antiproton Source.
http://www-ad.fnal.gov/public/antiproton.html

[12] Fermilab Tevatron Operations Rookie Books. *Concepts Rookie Book.*
http://www-bdnew.fnal.gov/operations/
rookie_books/rbooks.html

[13] Fermilab Tevatron Operations Rookie Books. *Linac Rookie Book.*
http://www-bdnew.fnal.gov/operations/
rookie_books/rbooks.html

[14] Tevatron Operations Rookie Books. *Antiproton Source Rookie Book.*
http://www-bdnew.fnal.gov/operations/
rookie_books/rbooks.html

[15] Fermilab Tevatron Operations Rookie Books. *Tevatron Rookie Book.*
http://www-bdnew.fnal.gov/operations/
rookie_books/rbooks.html

[16] CDF Collaboration. *CDF II Detector Technical Design Report.*
FERMILAB-PUB-96/390-E

[17] Feynman, Richard. (1997) *Six Not-So-Easy Pieces.* Basic Books.

[18] About Fermilab. *Fermilab FAQ.*
http://www.fnal.gov/pub/about/faqs/index.html

[19] Nelson, Timothy for the CDF Collaboration. *The CDF Layer 00 Detector.* Int.J.Mod.Phys A16S1C:1091-1093, 2001

[20] Vapnik, V N. *The Nature of Statistical Learning Theory* New York: Springer Verlag, 1995

[21] Pitts, Kevin. *The CDF Central Outer Tracker.* CDF Note 3944, (1996)

[22] Giagu, Stefano. *The CDFII Time-of-Flight Detector and Impact on Beauty Flavor Tagging.* CDF Note 6113, (2002)

[23] Levy, Stephen. *Tau Identification at the Tevatron.* Proceedings of the 1st Hadron Collider Physics Symposium, Les Diablerets, Switzerland, July 4-9, 2005

[24] PDF Systematic Procedures for CDF.
`http://www-cdf.fnal.gov/internal/physics/joint_physics/instructions/PDFUncertainties/pdf.html`

[25] Cacciari, M., *et al. The $t\bar{t}$ cross-section at 1.8 TeV and 1.96 TeV: A study of the systematics due to parton densities and scale dependence.* JHEP 04 (2004) 068, hep-ph/0303085.

[26] Cacciari, M., *et al. Updated predictions for the total production cross sections of top and of heavier quark pairs at the Tevatron and at the LHC.* arXiv:0804.2800v3.

[27] Kidonakis, N. & Vogt, R. *Top quark production at the Tevatron at NNLO.* Eur. Phys. J. C33 (2004) S466-S468, hep-ph/0309045.

[28] Kidonakis, N. & Vogt, R. *The theoretical top cross section at the Tevatron and the LHC.* arXiv:0805.3844v1.

[29] Keerthi, S.S., *et al. Improvements to Platt's SMO Algorithm for SVM Classifier Design.*

[30] Platt, J.C., *Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines.*

[31] Arguin, J., *et al. Generic Jet Energy Corrections for Run II data used for the Winter Conferences.* CDF Note 6280, (2003)

[32] Bhatti, A., *et al. Determination of the Jet Energy Scale at the Collider Detector at Fermilab.* CDF Note 7543, (2005)

[33] Bhatti, A., *et al. Absolute Corrections and their Systematic Uncertainties.* CDF Note 7450, (2005)

[34] Wagner, R.G., *Understanding and using L_{SHR}.* CDF Note 6249.

[35] Nodulman, Larry. *Central Electron Issues for Summer '03.* CDF Note 5572, (2003)

[36] Taffard, Anyes. *Run II Cosmic Ray Tagger.* CDF Note 6100, (2003)