1 Summary

The purpose of this project was to determine a model for classifying the presence of a Higgs boson from background noise given a set of detector measurements. The data used for modeling was taken from the UCI Machine Learning Depository [2], which were originally produced in Reference [1]. The large data set was sampled and analyzed, and features were separated into 2 sets. 10 machine learning models were fit to the 2 feature sets and their corresponding principle components. The model with the best results was a 3-layer fully-connected deep learning model. All models including the final model produced poor results, due predominantly to the well-mixed nature of the given data.

2 Methods

All data processing, analysis, and modeling was performed with the R statistical coding language, with the following code packages:

`ggplot2` for plotting functions
`GGally` for correlation plotting functions
`ROCR` for ROC plotting
`gridExtra` for plot arrangement
`dplyr` for data manipulation functions
`tidyr` for data wrangling functions
`caTools` for AUC calculations
`caret` for machine learning functions
`H2O` for deep neural network functions

All R code will be provided in a separate file, ”STAT724_cody_gilbert_exam.R.” The code produces output files in the form of ”*Output.txt” files that contain summary outputs of the models and test results, and ”*.pdf” files which contain additional plots. A large number of plots were produced for each model, and plots not related to the final model are not included within this report.

3 Data Sourcing, Screening, and Analysis

3.1 Data Sources

The data used in this calculation were taken from the UCI Machine Learning Depository [2], which were originally produced in Reference [1]. This data contains simulations of particle accelerator detectors, which either have a class of 1 for the presence of the Higgs boson, or 0 for the simulation of background noise [2]. The data originally contained 11,000,000 observations of 28 features: 21 are lower-tier simulated detector signals and 7 are parameters derived from the lower-tier features. To reduce the scale of the project, the original data set was sampled to obtain 50,000 observations on which the classification models will be run. The final neural network model used 200,000 observations.

3.2 Data Analysis

The sampled data is contained in the ”HIGGSSample.csv” file. The `Class` feature was reformulated to be ”H” (Higgs) for class 1, and ”B” (Background) for class 0. Appendix A shows R `summary` output and exploratory plots created from the sampled data. The summary output indicates no missing data, and a balanced number of classes.

The data was separated into 2 parts: a set of derived data features and a set of lower-tier features. The total set of features were not analyzed together, as the derived features are by definition collinear with the lower-tier features. Both data sets will be analyzed with the same set of models, and the features with the highest predictive capability will be used.

The correlation plot of a 1000 observation sample of the derived data is shown in Figure 3. The density plots show that the data are right-skewed with a number of features showing significant collinearity. The densities of the separate classes are shown on the right side of Figure 4. These densities show that the background cases and the Higgs cases are well mixed within the bulk of the data.

A correlation plot of the lower-tier features was not produced, as it would require a $21 * 21 = 441$ facet plot which would be difficult to interpret. Manual assessment of the correlation matrix indicated little correlation between features. A density plot with separate classes was produced from the 1000 observation sample, and is shown on the left side of Figure 4. The density plots again show that the classes are well mixed, with distributions that are mostly symmetric.

The derived data show significant collinearity, which can be eliminated by modeling on the principle components (PCs). Although the lower-tier data does not show a significant level of collinearity, the PCs were calculated for completeness. The PCs were calculated using the R `prcomp` function. Biplots and class-separated density plots for the PCs of the derived data are shown in Figure 5, and for the lower-tier data are shown in Figure 6. The plots do not indicate a significant improvement in data separation, however the PCs will be modeled to determine if they improve accuracy.

4 Modeling

The following classification models were considered:

1. Logistic
2. Linear Discriminant Analysis (LDA)
3. Quadratic Discriminant Analysis (QDA)
4. K-Nearest Neighbor (KNN)
5. Random Forest
6. Boosted Forest
7. Support Vector Machines (SVMs)
8. Neural Network

The `caret` package was used to select a 75% training set and a 25% test set. A common set of 5 cross-validation folds were used across models to allow for model comparison.

4.1 Variable Selection

For each data set, derived features, lower-tier features, and the PCs for both, variable selection was performed using the `caret` package `rfe` function for Recursive Feature Elimination (RFE). This algorithm used a random forest model to perform RFE sub-setting, which identified as important all 7 derived features and their PCs, and 16 of the lower-tier features and their PCs. Because all 7 PCs of the derived data were indicated as important, the number was reduced to 5 in order to account for over 95% of the total variance.

It is assumed that the RFE algorithm using a random forest model can be generalized to account for all considered models, as the algorithm will provide a first-order indication of variable importance. This assumption can be validated by re-running the RFE algorithm with each model used, however this was not performed in this project due to computing time limitations. The highest accuracy models perform cross-validation to tune regularization parameters, therefore the likelihood of overfitting is minimal.

4.2 Logistic Model

A logistic model was prepared for all 4 feature sets with the features identified in Section 4.1. Model fitting and prediction was performed using the `caret` package's cross-validation wrapper to the `glm` function.

Features were assumed to have only a simple linear formulation with no transformations or interactions. This simple model is justified because the class-separated density plots in Figure 4 show that the data is well-mixed and a logistic model is not expected to perform with a high-level of accuracy. It is expected that this model will form a baseline level of prediction on which subsequent models can be compared. Table 1 shows the best results for each data feature set.

Table 1: Logistic Model Validation Results

Feature Set	**AUC**	**Sensitivity**	**Specificity**
Derived	0.648	0.438	0.792
Lower-Tier	0.582	0.396	0.702
Derived PCs	0.566	0.349	0.826
Lower-Tier PCs	0.583	0.394	0.703

4.3 LDA Model

An LDA model was prepared for all 4 feature sets with the features identified in Section 4.1. Model fitting and prediction was performed using the `caret` package's cross-validation wrapper to the `lda` function. Table 2 shows the best results for each data feature set.

Table 2: LDA Model Validation Results

Feature Set	**AUC**	**Sensitivity**	**Specificity**
Derived	0.647	0.417	0.808
Lower-Tier	0.582	0.395	0.703
Derived PCs	0.567	0.336	0.836
Lower-Tier PCs	0.583	0.393	0.704

4.4 QDA Model

A QDA model was prepared for all 4 feature sets with the features identified in Section 4.1. Model fitting and prediction was performed using the `caret` package's cross-validation wrapper to the `qda` function. Table 3 shows the best results for each data feature set.

Table 3: QDA Model Validation Results

Feature Set	**AUC**	**Sensitivity**	**Specificity**
Derived	0.693	0.358	0.856
Lower-Tier	0.631	0.449	0.724
Derived PCs	0.647	0.309	0.857
Lower-Tier PCs	0.631	0.446	0.727

4.5 KNN Model

A KNN classification model was prepared for all 4 feature sets with the features identified in Section 4.1. Model fitting and prediction was performed using the `caret` package's cross-validation wrapper to the `knn` function. The `caret` package performed 5-fold cross validation to tune the k parameter over values of 1, 5, 10, 20, 50, and 100. Table 4 shows the best results for each data feature set.

Table 4: KNN Model Validation Results

Feature Set	**AUC**	**Sensitivity**	**Specificity**	**k**
Derived	0.746	0.600	0.741	50
Lower-Tier	0.598	0.333	0.789	50
Derived PCs	0.709	0.531	0.758	100
Lower-Tier PCs	0.635	0.386	0.783	50

4.6 Random Forest Model

A random forest classification model was prepared for all 4 feature sets with the features identified in Section 4.1. Model fitting and prediction was performed using the `caret` package's cross-validation wrapper to the `ranger` function. The Gini index was used to determine node purity. The `caret` package performed 5-fold cross validation to tune the following parameters:

`mtry` (bootstrapped parameters): 2, $sqrt(p)$, p, where p are the features
`min.node.size`: 10, 20, 50, 100

Table 5 shows the best results for each data feature set.

Table 5: Random Forest Model Validation Results

Feature Set	**AUC**	**Sensitivity**	**Specificity**	**mtry**	**min.node.size**
Derived	0.762	0.649	0.725	3	50
Lower-Tier	0.658	0.509	0.704	4	10
Derived PCs	0.708	0.575	0.719	3	100
Lower-Tier PCs	0.664	0.497	0.724	2	10

4.7 Boosted Tree Model

A boosted tree classification model was prepared for all 4 feature sets with the features identified in Section 4.1. Model fitting and prediction was performed using the `caret` package's cross-validation wrapper to the `gbm` function. The `caret` package performed 5-fold cross validation to tune the following parameters:

`n.tree` (number of trees): 10, 50, 100, 500, 700
`shrinkage`: 0, 0.01, 0.1
`n.minobsinnode` (minimum obs. in node): 10, 20
`interaction.depth`: 1, 5

Table 6 shows the best results for each data feature set.

Table 6: Boosted Model Validation Results

Feature Set	**AUC**	**Sensitivity**	**Specificity**	**n.trees**	**int.depth**	**shrinkage**	**m.obs**
Derived	0.760	0.646	0.725	700	5	0.01	20
Lower-Tier	0.639	0.438	0.736	700	5	0.01	20
Derived PCs	0.696	0.526	0.746	100	5	0.1	20
Lower-Tier PCs	0.642	0.521	0.680	500	5	0.1	20

4.8 Support Vector Machine Models

Three SVM models discussed in class were considered: linear kernel, polynomial kernel, and radial kernel. Due to computing time restrictions, the polynomial kernel could not be used to model the final data sets. Based on the well-mixed nature of the data and the results of the Logistic, LDA, and QDA models, the polynomial kernel SVM model is not expected to provide dramatically improved results.

4.8.1 Linear Kernel

An SVM model with a linear kernal was prepared for all 4 feature sets with the features identified in Section 4.1. Model fitting and prediction was performed using the `caret` package's cross-validation wrapper to the `svmLinear` function. The `caret` package performed 5-fold cross validation to tune the cost parameter over values 10, 50, 75, and 100. Table 7 shows the best results for each data feature set.

Table 7: SVM Linear Model Validation Results

Feature Set	**AUC**	**Sensitivity**	**Specificity**	**Cost**
Derived	0.647	0.425	0.801	50
Lower-Tier	0.580	0.359	0.734	10
Derived PCs	0.563	0.337	0.834	50
Lower-Tier PCs	0.581	0.356	0.735	100

4.8.2 Radial Kernel

An SVM model with a radial kernal was prepared for all 4 feature sets with the features identified in Section 4.1. Model fitting and prediction was performed using the `caret` package's cross-validation wrapper to the `svmRadial` function. The `caret` package performed 5-fold cross validation to tune the following parameters:

`c` (cost): 1, 10, 50, 100
`sigma`: 0.001, 0.01, 1, 5, 10

Table 8 shows the best results for each data feature set.

Table 8: SVM Linear Model Validation Results

Feature Set	**AUC**	**Sensitivity**	**Specificity**	**Cost**	**Sigma**
Derived	0.748	0.644	0.718	1	1
Lower-Tier	0.653	0.471	0.736	100	0.01
Derived PCs	0.705	0.593	0.707	1	1
Lower-Tier PCs	0.654	0.466	0.744	100	0.01

4.9 Neural Network Models

4.9.1 Shallow Network

A single hidden layer, fully-connected neural network classification model was prepared for all 4 feature sets with the features identified in Section 4.1. Model fitting and prediction was performed using the `caret` package's cross-validation wrapper to the `nnet` function. The `caret` package performed 5-fold cross validation to tune the following parameters:

`size` (hidden layer nodes): 10, 50, 100
`decay` (regularization parameter): 0, 0.1, 1

Table 9 shows the best results for each data feature set.

Table 9: Shallow Neural Network Model Validation Results

Feature Set	**AUC**	**Sensitivity**	**Specificity**	**Size**	**decay**
Derived	0.757	0.636	0.735	50	0.1
Lower-Tier	0.639	0.526	0.670	50	1
Derived PCs	0.714	0.599	0.705	10	0.1
Lower-Tier PCs	0.650	0.546	0.668	50	1

4.9.2 Deep Network

A deep, fully-connected neural network classification model was prepared with the Derived feature set. The Derived feature set was used because it performed the best of the feature sets from the above models, and computing deep neural network models for all the sets would require an unacceptable amount of computation time. Model fitting and prediction was performed using the `H2O` package. Because the `H2O` can fit a model faster than `caret` package, the sample size was increased from 50,000 to 200,000. A separate training and test set were sampled using the h2o.splitFrame function. The H2O h2o.grid function used a random search with 5-fold cross validation to tune the following hyperparameters:

hidden[1] (hidden layers): 1, 2, 3
hidden[2] (hidden layer nodes): 10, 25, 50, 100, 200
input_dropout_ratio: 0, 0.5
rate: 0.01, 0.02

The grid tuning used a sample of 50,000 of the training observations to speed computation. After tuning, a model was fit using the entire training set with the best parameters determined from the h2o.grid models. This model used 3 layers of 50 nodes, a dropout ratio of 0, and a learning rate of 0.02. All other model parameters used the default options for the h2o.deeplearning model. This model produced a 5-fold cross validation AUC of 0.785, and an accuracy of 0.707. The model test AUC was 0.780, test accuracy was 0.704, and the model produced the following test confusion matrix:

	Test Higgs Data	
Predicted	B	H
B	10798	12780
H	3149	23282

An ROC plot of the test data is shown in Figure 1.

5 Conclusion

A graphical overview of the model AUCs is shown in Figure 2. The model with the best cross validation AUC and accuracy was the Deep Neural Network (Section 4.9.2) model assembled with the H2O h2o.deeplearning function. The final test AUC was 0.780, the final test accuracy was 0.704. All of the models ultimately performed rather poorly, with the worse models being the linear decision boundary models that performed only slightly better than the null model. The well-mixed nature of the data lead to an improvement in accuracy for non-linear models, but ultimately the data is difficult to separate.

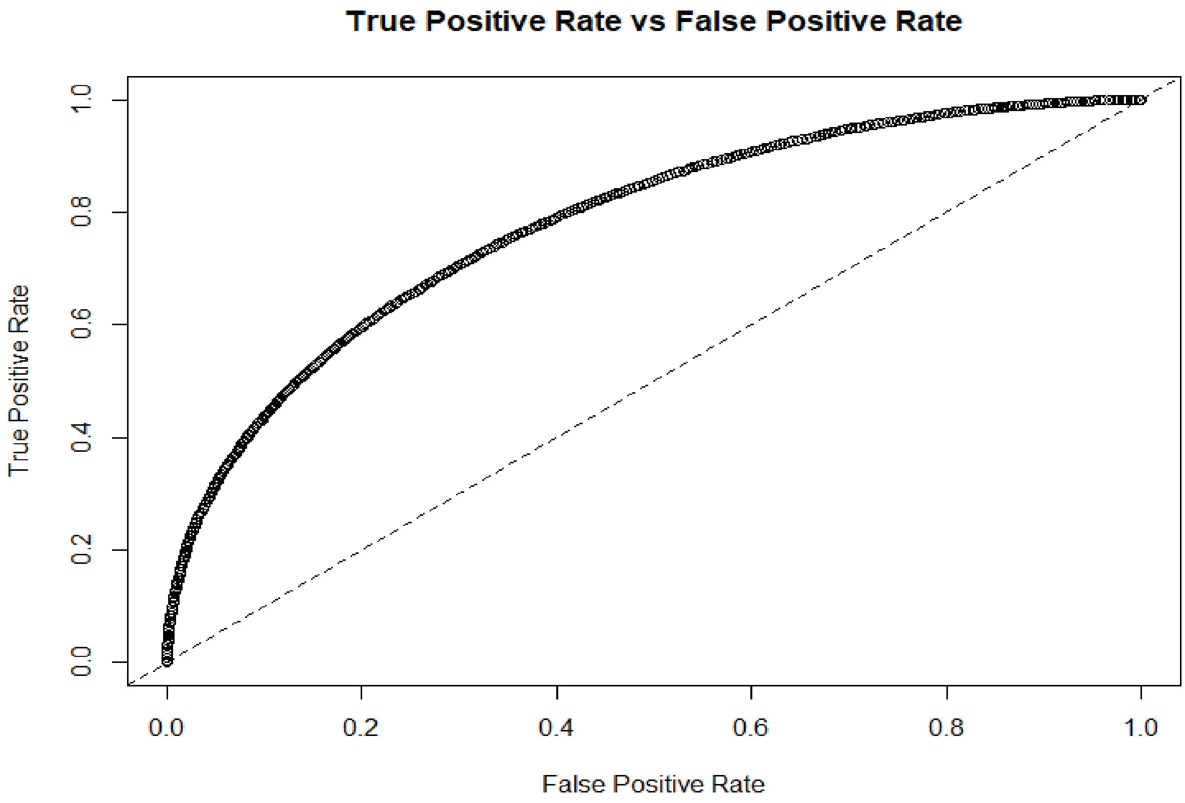

Figure 1: Deep Neural Network Model Test ROC Curve

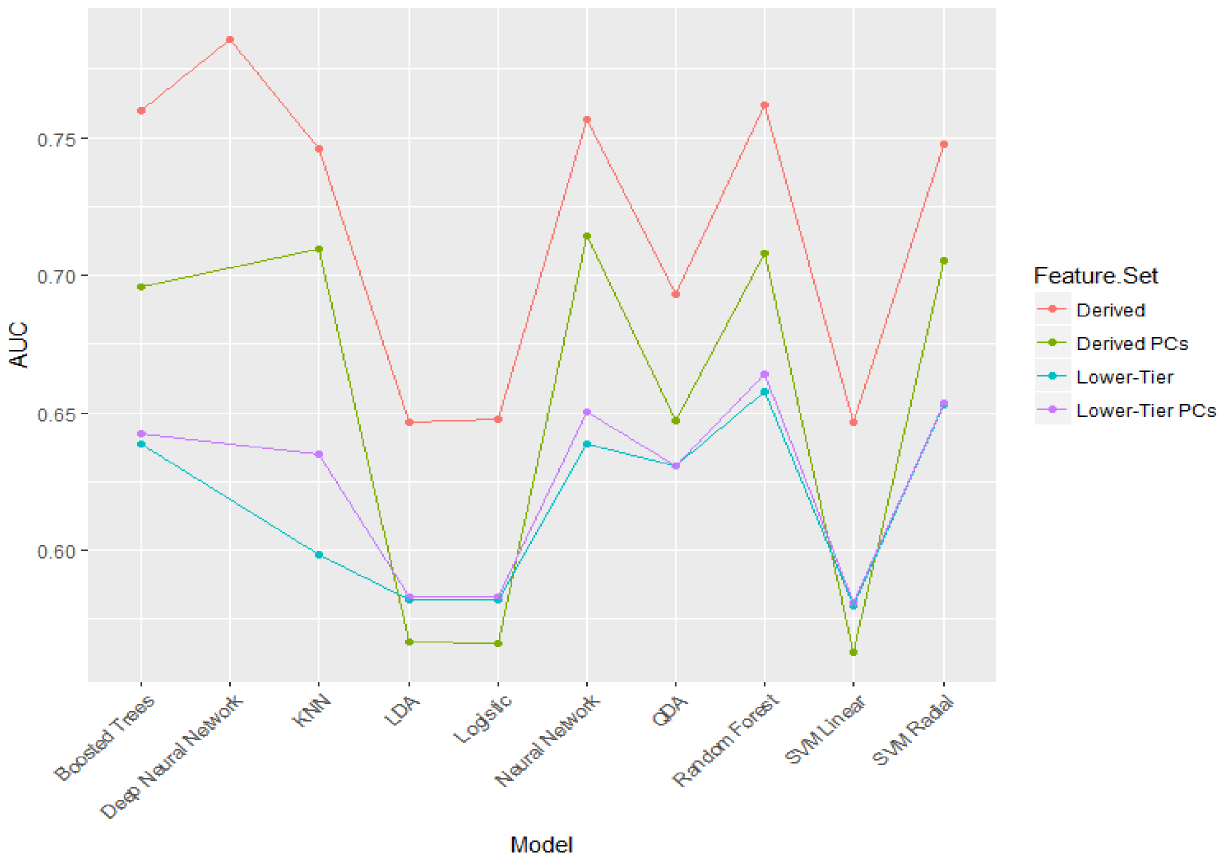

Figure 2: Final 5-fold training AUC per Model. Note that the Deep Neural Network model was only run with the Derived Feature set, with a larger data sample.

References

[1] Baldi, P. Sadowski, P., Whiteson, D. Searching for Exotic Particles in High-Energy Physics with Deep Learning (2014). `https://arxiv.org/pdf/1402.4735.pdf` Irvine, CA: University of California, School of Information and Computer Science.

[2] Dua, D. and Karra Taniskidou, E. (2014). UCI Machine Learning Repository, `http://archive.ics.uci.edu/ml/datasets/HIGGS`. Irvine, CA: University of California, School of Information and Computer Science.

A Appendix: Higgs Data Summaries

The first feature column contains the Class feature, ”B” for Background noise and ”H” for a Higgs particle. The following 21 features are detector simulations, and the final 7 are values derived from the detector outputs.

```
 Class        lepton.pT        lepton.eta          lepton.phi
 B:23435   Min.   :0.2747   Min.   :-2.434976   Min.   :-1.7425083
 H:26565   1st Qu.:0.5929   1st Qu.:-0.745140   1st Qu.:-0.8719308
           Median :0.8554   Median :-0.009794   Median : 0.0054103
           Mean   :0.9912   Mean   :-0.006811   Mean   :-0.0007317
           3rd Qu.:1.2346   3rd Qu.: 0.732370   3rd Qu.: 0.8732134
           Max.   :7.0003   Max.   : 2.431946   Max.   : 1.7432359
 missing.energy.magnitude missing.energy.phi     jet.1.pt        jet.1.eta
 Min.   :0.003158         Min.   :-1.743944   Min.   :0.1400   Min.   :-2.968735
 1st Qu.:0.572091         1st Qu.:-0.881750   1st Qu.:0.6773   1st Qu.:-0.696157
 Median :0.888489         Median :-0.013527   Median :0.8932   Median :-0.002006
 Mean   :0.993825         Mean   :-0.008596   Mean   :0.9876   Mean   :-0.006254
 3rd Qu.:1.291927         3rd Qu.: 0.863734   3rd Qu.:1.1700   3rd Qu.: 0.687194
 Max.   :6.582300         Max.   : 1.743102   Max.   :7.0647   Max.   : 2.969674
   jet.1.phi          jet.1.b.tag       jet.2.pt        jet.2.eta
 Min.   :-1.741237   Min.   :0.0000   Min.   :0.1890   Min.   :-2.913089
 1st Qu.:-0.869759   1st Qu.:0.0000   1st Qu.:0.6586   1st Qu.:-0.694472
 Median :-0.011307   Median :1.0865   Median :0.8895   Median : 0.002003
 Mean   :-0.002803   Mean   :0.9955   Mean   :0.9917   Mean   : 0.001438
 3rd Qu.: 0.869421   3rd Qu.:2.1731   3rd Qu.:1.2000   3rd Qu.: 0.695564
 Max.   : 1.741454   Max.   :2.1731   Max.   :8.2802   Max.   : 2.912238
   jet.2.phi           jet.2.b.tag      jet.3.pt        jet.3.eta
 Min.   :-1.7423717   Min.   :0.000   Min.   :0.2636   Min.   :-2.729663
 1st Qu.:-0.8657405   1st Qu.:0.000   1st Qu.:0.6549   1st Qu.:-0.697078
 Median :-0.0002033   Median :1.107   Median :0.8978   Median : 0.001993
 Mean   : 0.0007803   Mean   :1.006   Mean   :0.9928   Mean   : 0.002306
 3rd Qu.: 0.8698727   3rd Qu.:2.215   3rd Qu.:1.2213   3rd Qu.: 0.697423
 Max.   : 1.7431748   Max.   :2.215   Max.   :8.5099   Max.   : 2.730009
   jet.3.phi          jet.3.b.tag      jet.4.pt        jet.4.eta
 Min.   :-1.742069   Min.   :0.000   Min.   :0.3654   Min.   :-2.497265
 1st Qu.:-0.861842   1st Qu.:0.000   1st Qu.:0.6173   1st Qu.:-0.721686
 Median : 0.003789   Median :0.000   Median :0.8721   Median :-0.007956
 Mean   : 0.003523   Mean   :1.006   Mean   :0.9872   Mean   :-0.005375
 3rd Qu.: 0.876942   3rd Qu.:2.548   3rd Qu.:1.2214   3rd Qu.: 0.714102
 Max.   : 1.742884   Max.   :2.548   Max.   :7.5120   Max.   : 2.498009
   jet.4.phi          jet.4.b.tag           m_jj              m_jjj            m_lv
 Min.   :-1.742691   Min.   :0.0000   Min.   : 0.1255   Min.   :0.3420   Min.   :0.2668
 1st Qu.:-0.878138   1st Qu.:0.0000   1st Qu.: 0.7907   1st Qu.:0.8461   1st Qu.:0.9858
 Median :-0.005259   Median :0.0000   Median : 0.8949   Median :0.9501   Median :0.9897
 Mean   :-0.003756   Mean   :0.9869   Mean   : 1.0319   Mean   :1.0221   Mean   :1.0500
 3rd Qu.: 0.863975   3rd Qu.:3.1020   3rd Qu.: 1.0254   3rd Qu.:1.0825   3rd Qu.:1.0196
 Max.   : 1.743372   Max.   :3.1020   Max.   :16.6016   Max.   :8.9401   Max.   :3.9318
```

```
     m_jlv             m_bb              m_wbb             m_wwbb
 Min.   :0.3199   Min.   : 0.06639   Min.   :0.3034   Min.   :0.3509
 1st Qu.:0.7681   1st Qu.: 0.67453   1st Qu.:0.8199   1st Qu.:0.7696
 Median :0.9175   Median : 0.87469   Median :0.9471   Median :0.8709
 Mean   :1.0112   Mean   : 0.97412   Mean   :1.0320   Mean   :0.9588
 3rd Qu.:1.1417   3rd Qu.: 1.14119   3rd Qu.:1.1372   3rd Qu.:1.0584
 Max.   :7.4426   Max.   :11.99418   Max.   :6.4013   Max.   :4.8353
```

The 7 derived features are m_jj, m_jjj,m_lv, m_jlv, m_bb, m_wbb, and m_wwbb, and the remaining features are the lower-tier features.

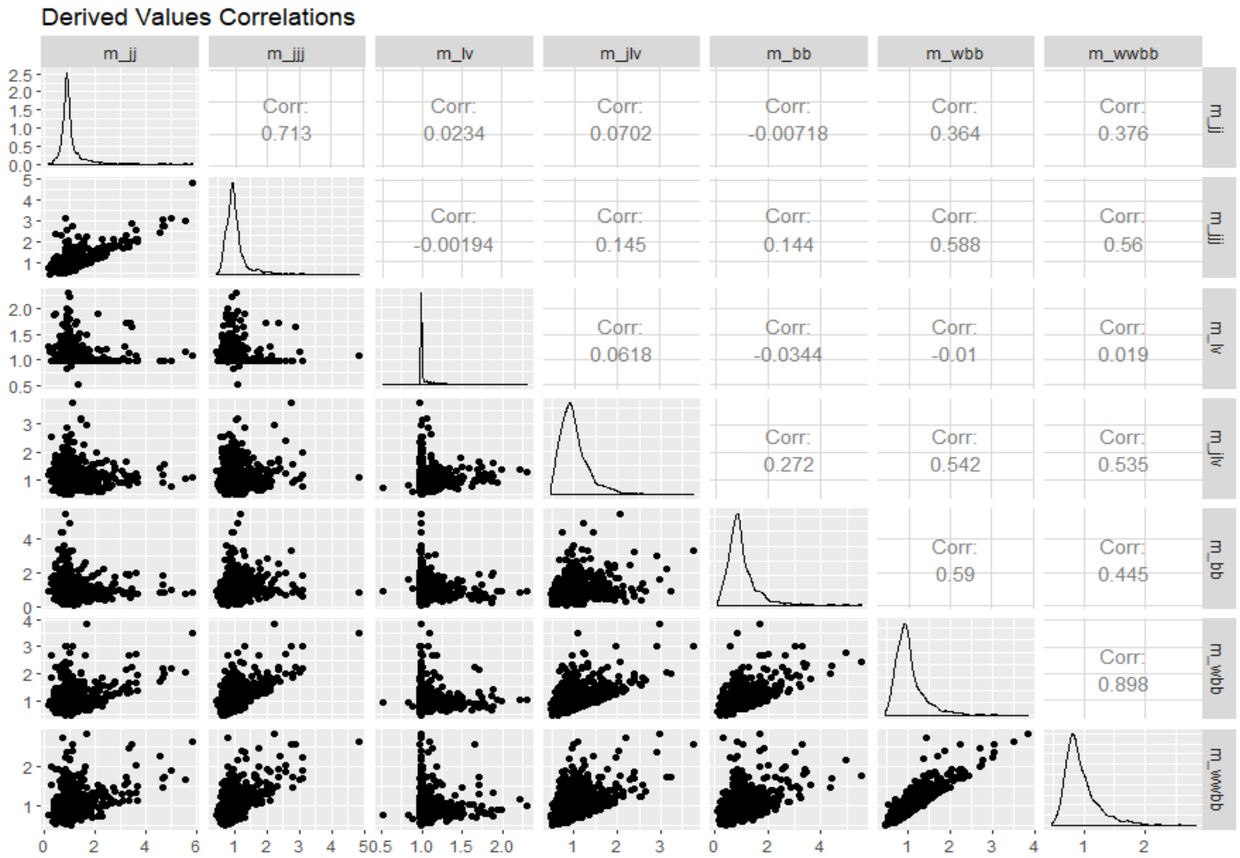

Figure 3: Derived Features Correlation plot. Note that this is a plot of a 1,000 observation sample.

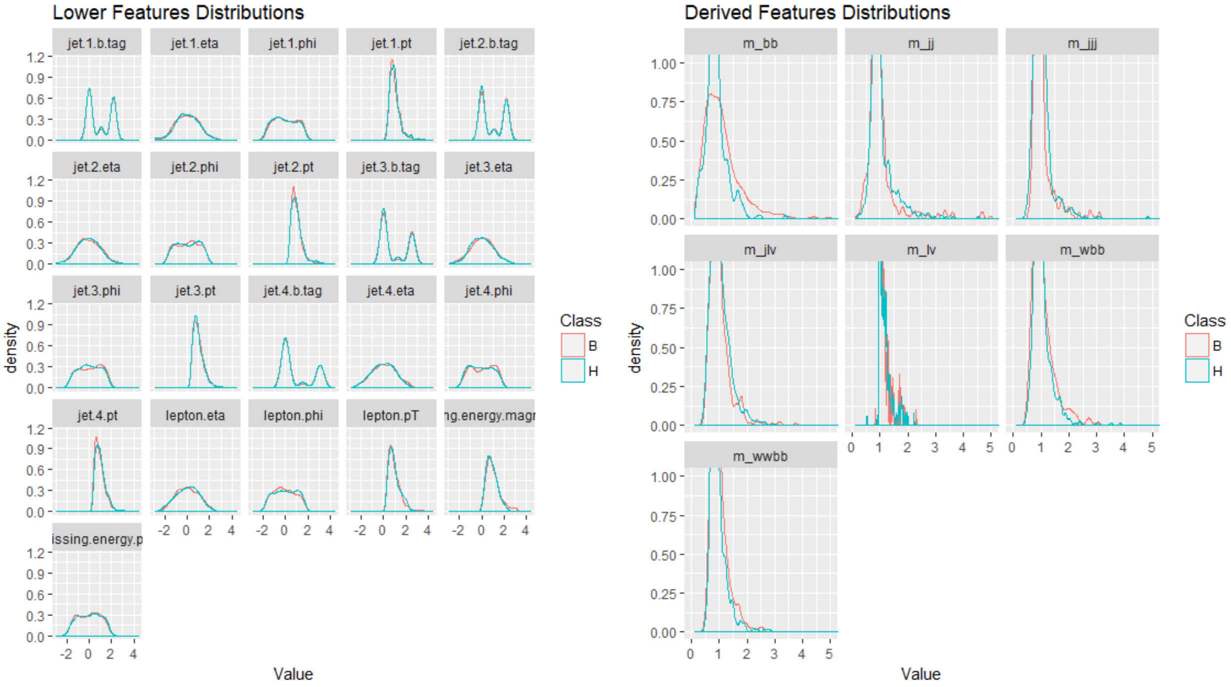

Figure 4: Density plots of Lower and Derived Features. Note that this is a plot of a 1,000 observation sample. The plotting windows of the Derived features were adjusted to show the distribution of the majority of data.

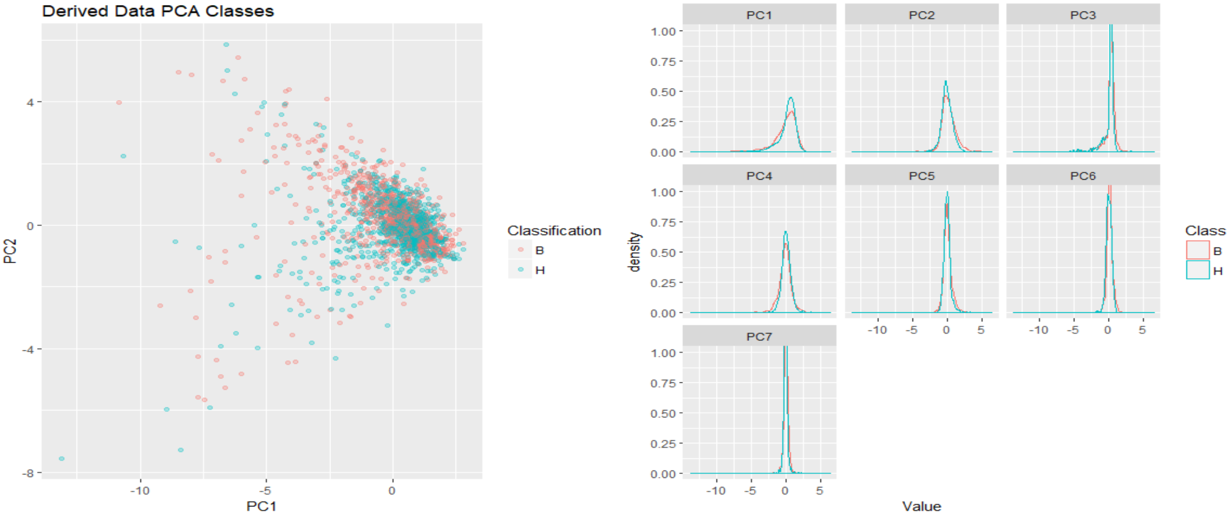

Figure 5: Biplot and Density plots of the Derived Features. Note that this is a plot of a 1,000 observation sample.

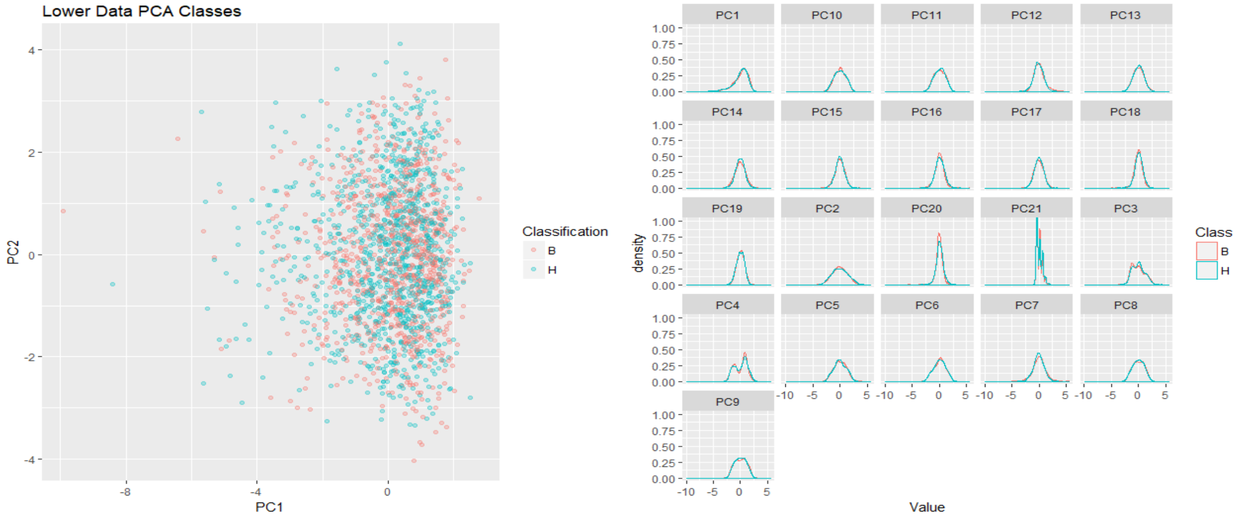

Figure 6: Biplot and Density plots of the Lower Features. Note that this is a plot of a 1,000 observation sample.