Summary: ML Analysis of Diabetic Patient Data and Associated Factors

- Public dataset of females at least 21 years old of Pima Indian Heritage
  - Note: may not be generalizable given specific population of interest
- 8 attributes, 1 explanatory variable → Had or did not have diabetes for 768 patients
  - Missing, unreasonable or flawed data excluded completely in univariate analysis
  - Improvement: multiple-chain imputation for inexplicable values
- Supervised Classification Learning (Split the data into 70% training, 30% testing)
  - Multivariate Logistic Regression
  - Naive Bayesian Model
  - K-Nearest Neighbors Model
  - **Random Forest Algorithm with Feature Selection -- most successful, ~91% accuracy**
- Conclusion:
  - Strongest indicative factors for onset of diabetes: insulin and glucose levels -- on the contrary, blood pressure and diabetes pedigree does not seem to have significance
  - Would be better to have longitudinal variables -- dataset limitation

Statistics Final Project: Diabetes

By: Lillian Yu and Sadhvi Narayanan

Why Diabetes?

Diabetes is a chronic disorder of carbohydrate metabolism involving insulin. Symptoms include elevated sugar in the urine and the blood, excessive urination, thirst, hunger, weakness, weight loss, and itching. (NHIS)

Diabetes is a major cause of blindness, kidney failure, heart attacks, stroke and lower limb amputation. Between 2000 and 2016, there was a 5% increase in premature mortality from diabetes. In 2019, diabetes was the ninth leading cause of death with an estimated 1.5 million deaths directly caused by diabetes. (WHO)

About the Dataset

https://www.kaggle.com/datasets/mathchi/diabetes-data-set

The diagnostic, binary-valued variable investigated is whether the patient shows signs of diabetes according to World Health Organization criteria (i.e., if the 2 hour post-load plasma glucose was at least 200 mg/dl at any survey examination or if found during routine medical care).

Selected from a larger database:

- Constraints
  - All patients are females at least 21 years old of Pima Indian Heritage
  - the population lives near Phoenix, Arizona, USA

Attributes

Sample Size: 768

Number of Attributes or Dependent/Response variables: 8 (quantitative)

1) Number of times pregnant
2) Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3) Diastolic blood pressure (mm Hg)
4) *Triceps skin fold thickness (mm)***
5) 2-Hour serum insulin (mu U/ml)
6) Body mass index (weight in kg/(height in m)^2)
7) Diabetes pedigree function
8) Age (years)

Explanatory/Class variable: (0 - tested negative for diabetes or 1 - tested positive for diabetes)

***Triceps and subscapular skinfold thicknesses provide an index of body fat and midarm muscle circumference provides a measure of muscle mass and helps determine a person's body composition and body fat percentage*

Sample of 15 subjects from the dataset

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
5	5	116	74	0	0	25.6	0.201	30	0
6	3	78	50	32	88	31.0	0.248	26	1
7	10	115	0	0	0	35.3	0.134	29	0
8	2	197	70	45	543	30.5	0.158	53	1
9	8	125	96	0	0	0.0	0.232	54	1
10	4	110	92	0	0	37.6	0.191	30	0
11	10	168	74	0	0	38.0	0.537	34	1
12	10	139	80	0	0	27.1	1.441	57	0
13	1	189	60	23	846	30.1	0.398	59	1
14	5	166	72	19	175	25.8	0.587	51	1

Analysis Steps

1. Filtering and Visualization of data
2. Inference: Determining association between attributes and outcome using Logistic Regression
3. Machine Learning: Naive Bayesian Model Training
4. Machine Learning: k-Nearest Neighbors Model Training
5. Machine Learning: Random Forest Model Training

Filtering and visualization of data

- Implausible Values:
  - Removal of zeros e.g. bmi or insulin levels
  - Removal of outliers as a whole from each attribute/category

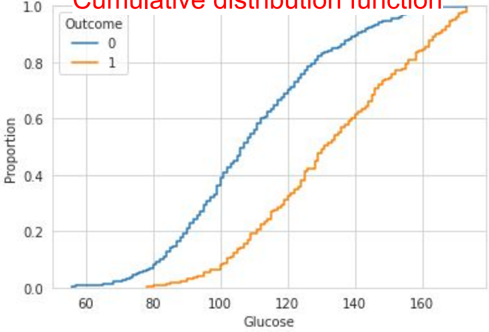
- Initially:
- Total number of positive diabetes (1): 268 individuals
- Total number of negative/no diabetes (0): 500 individuals

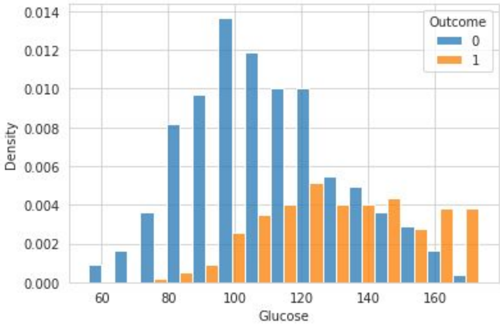
- Insulin and Skin Thickness had many missing values which may serve as a risk factor for inferences or associations

Distribution of Glucose and Outcome

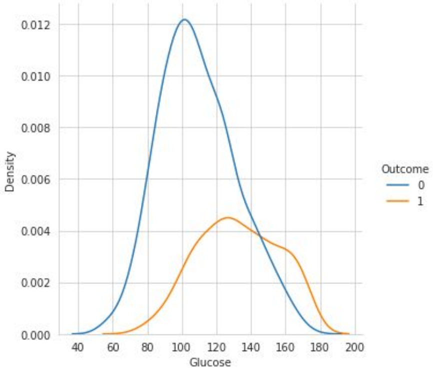
Since much of the data can not be described using a normal probability distribution, we will use ML to describe and analyze more complicated distributions

Cumulative distribution function

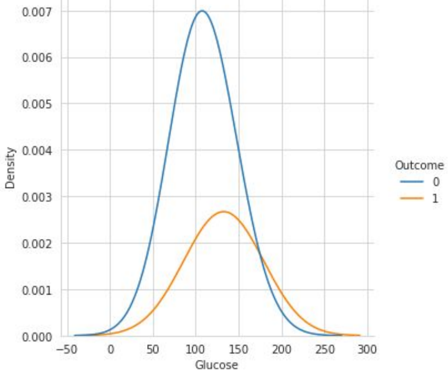




Probability density function

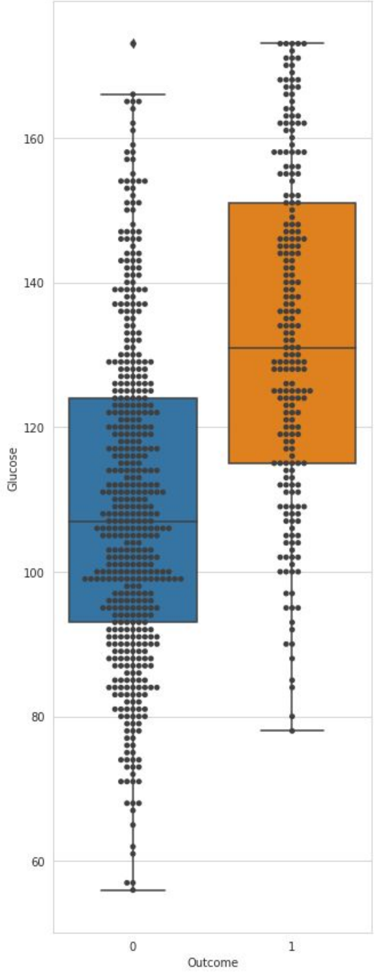


“Normalized” density function

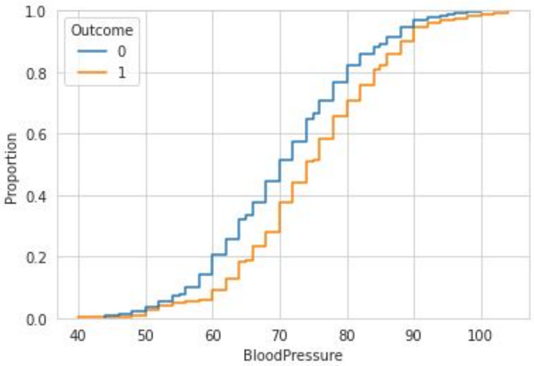


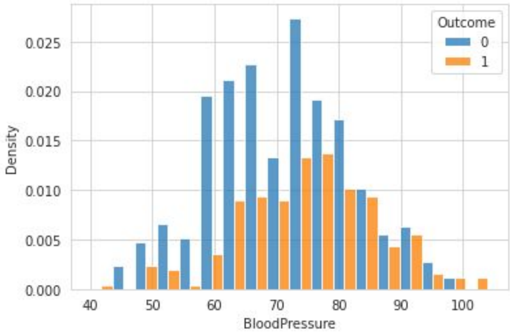
Positive Diabetes:
count 266.000000
mean 142.319549
std 29.599199
min 78.000000
25% 119.000000
50% 140.000000
75% 167.000000
max 199.000000

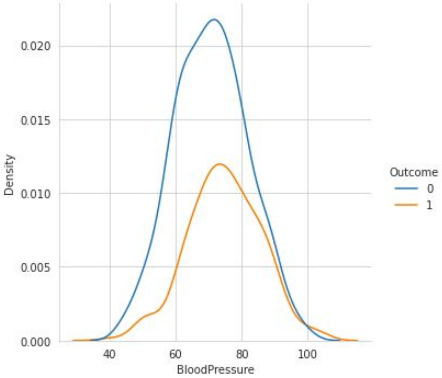
Negative Diabetes:
count 486.000000
mean 109.230453
std 22.326637
min 56.000000
25% 93.000000
50% 107.000000
75% 124.000000
max 173.000000

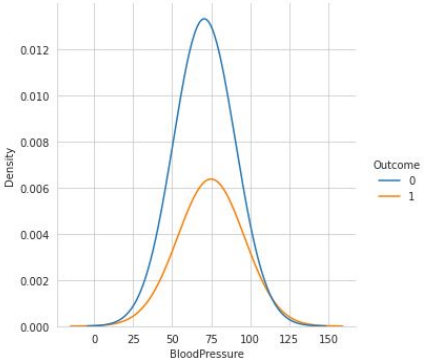


Distribution of Blood Pressure and Outcome







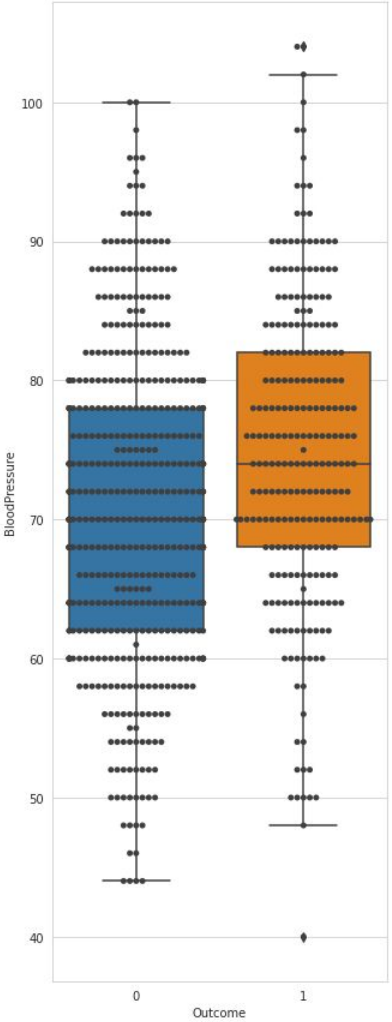


Positive Diabetes:

count	246.000000
mean	74.808943
std	11.055338
min	40.000000
25%	68.000000
50%	74.000000
75%	82.000000
max	104.000000

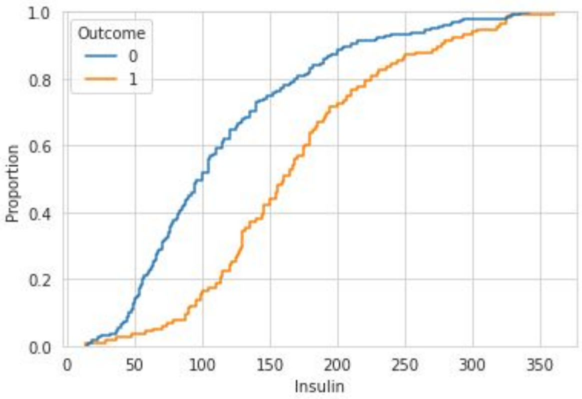
Negative Diabetes:

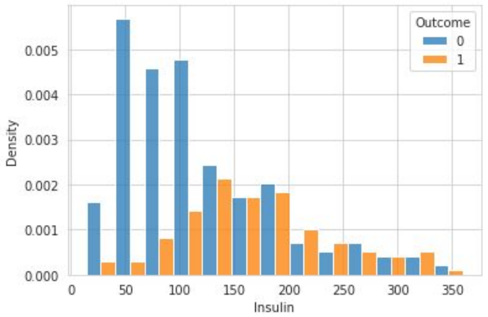
count	473.000000
mean	70.714588
std	11.088862
min	44.000000
25%	62.000000
50%	70.000000
75%	78.000000
max	100.000000

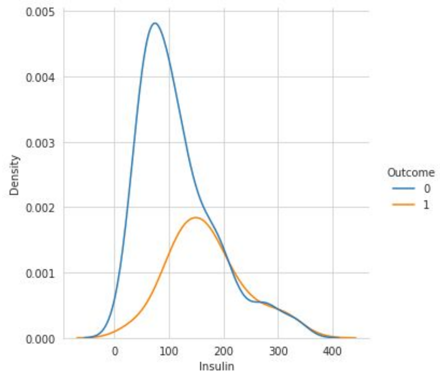


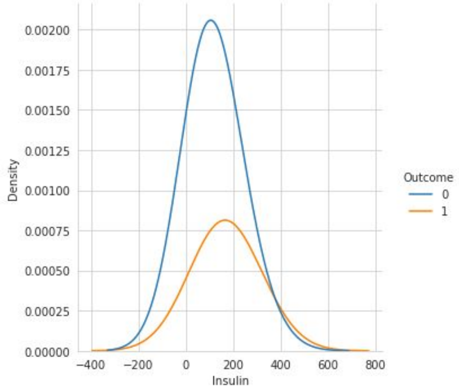
Distribution of Insulin and Outcome

**Flawed description of skewed distribution using normal distribution - original mode lost







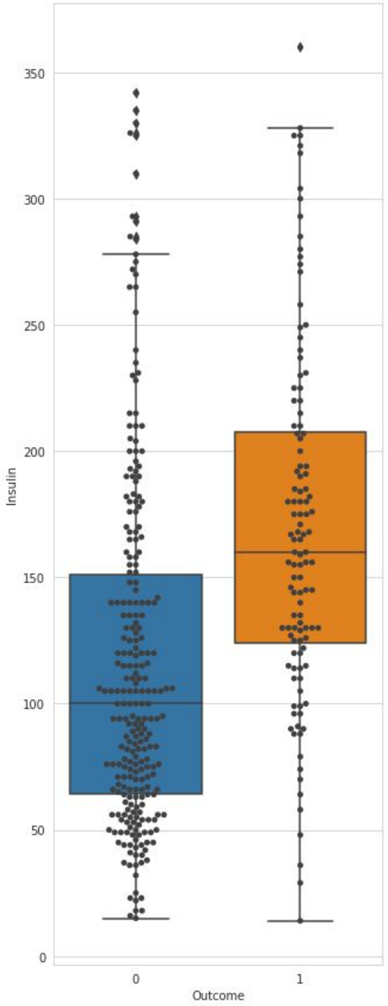


Positive Diabetes:

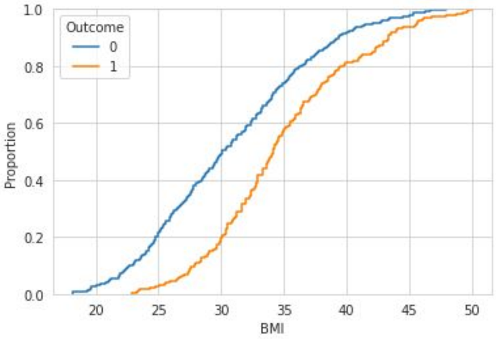
count	116.000000
mean	169.163793
std	70.789454
min	14.000000
25%	124.250000
50%	160.000000
75%	207.750000
max	360.000000

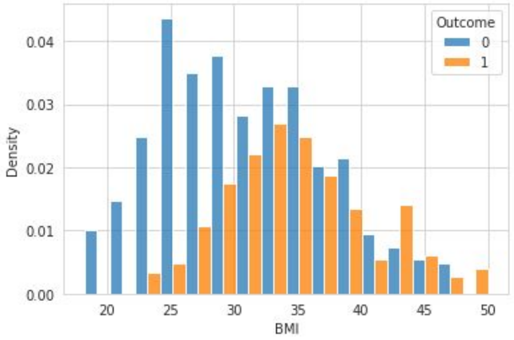
Negative Diabetes:

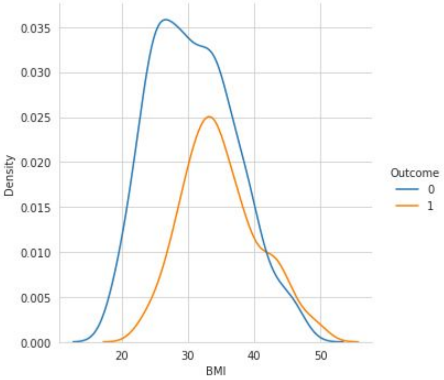
count	254.000000
mean	115.917323
std	69.844697
min	15.000000
25%	64.250000
50%	100.000000
75%	151.000000
max	342.000000

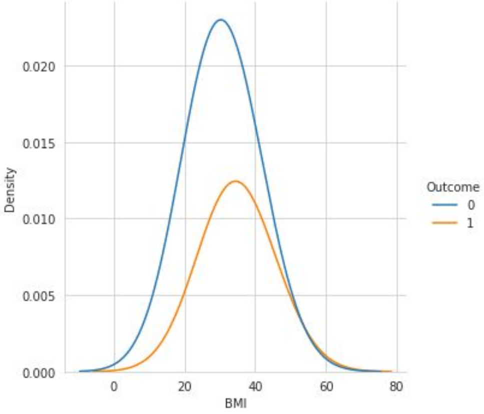


Distribution of BMI and Outcome







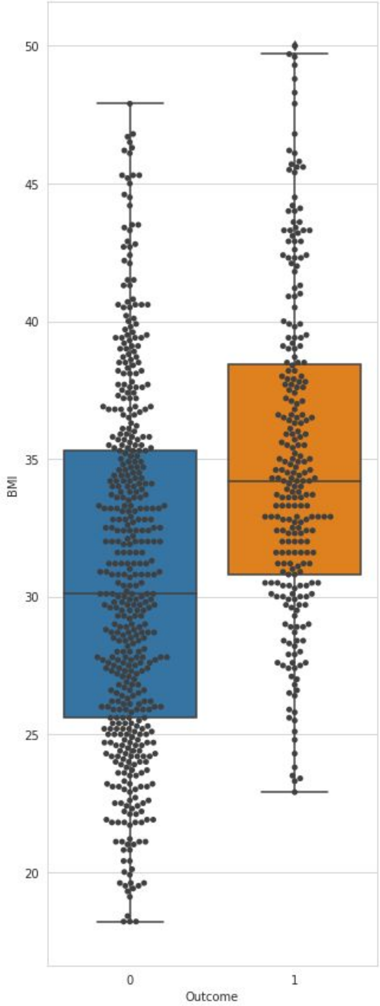


Positive Diabetes:

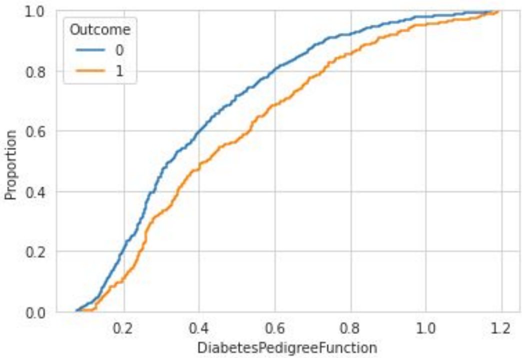
count	260.000000
mean	34.916538
std	5.782866
min	22.900000
25%	30.800000
50%	34.200000
75%	38.425000
max	50.000000

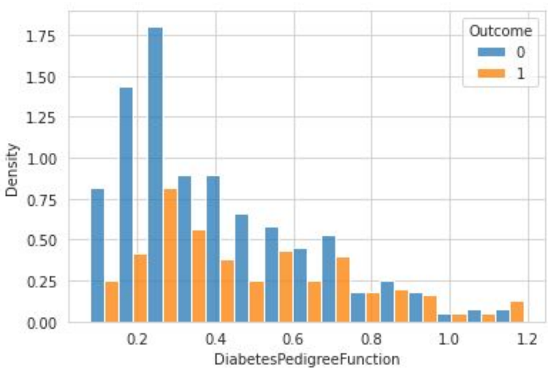
Negative Diabetes:

count	489.000000
mean	30.761759
std	6.390268
min	18.200000
25%	25.600000
50%	30.100000
75%	35.300000
max	47.900000



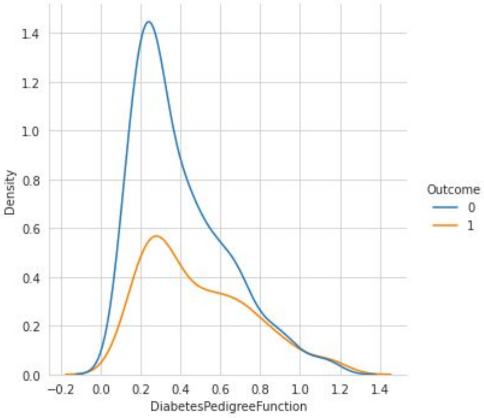
Distribution of Diabetes Pedigree Function and Outcome

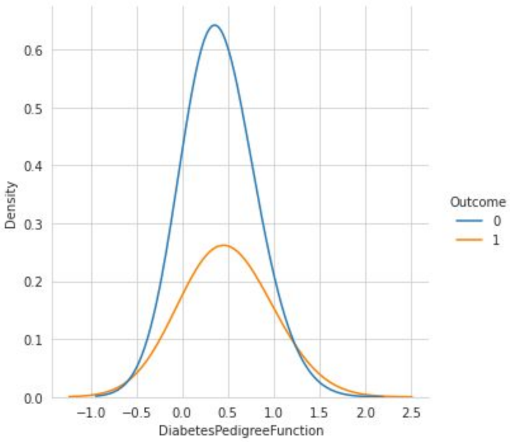


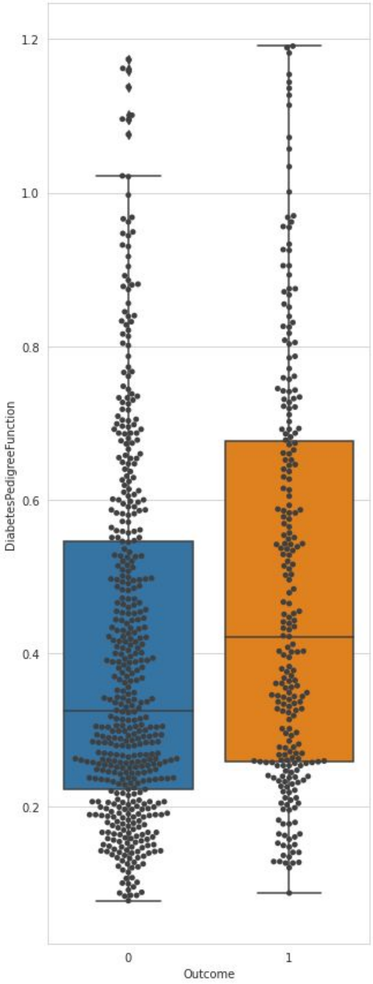


Positive Diabetes:
count 251.000000
mean 0.485713
std 0.266836
min 0.088000
25% 0.259500
50% 0.422000
75% 0.676000
max 1.191000

Negative Diabetes:
count 488.000000
mean 0.401090
std 0.235553
min 0.078000
25% 0.223000
50% 0.325000
75% 0.546250
max 1.174000

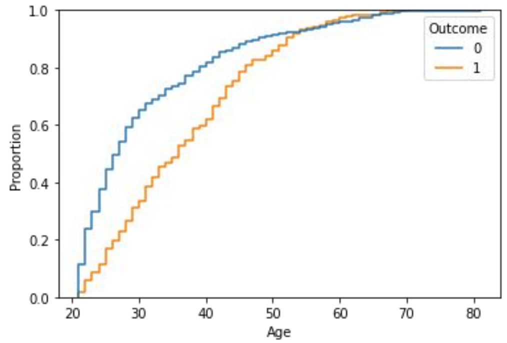


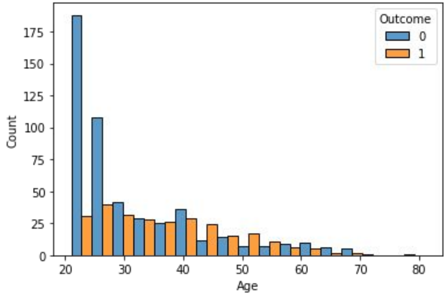


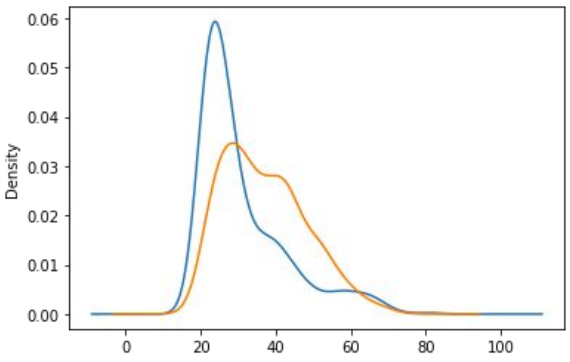


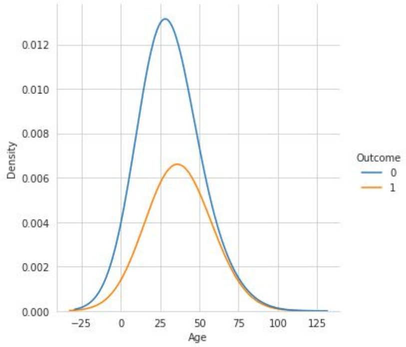
Distribution of Age and Outcome

Note: outliers were not removed for this specific dataset







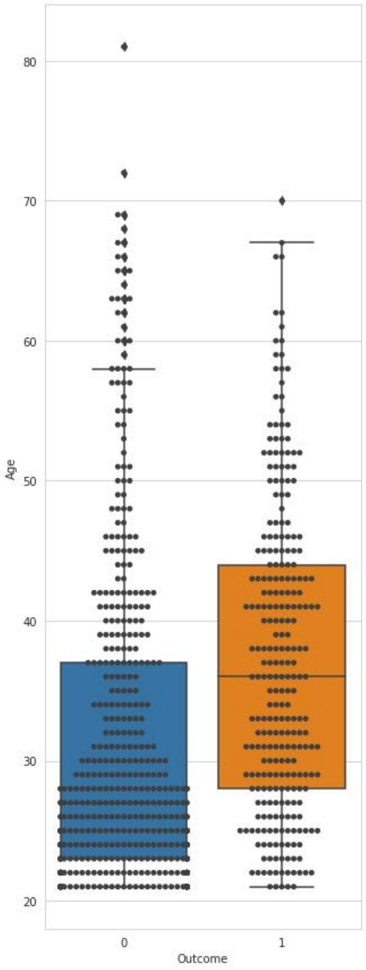


Positive Diabetes:

count	268.000000
mean	37.067164
std	10.968254
min	21.000000
25%	28.000000
50%	36.000000
75%	44.000000
max	70.000000

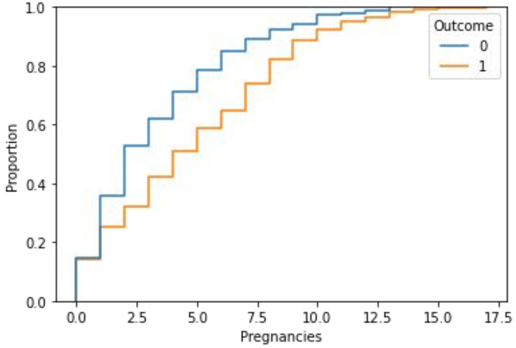
Negative Diabetes:

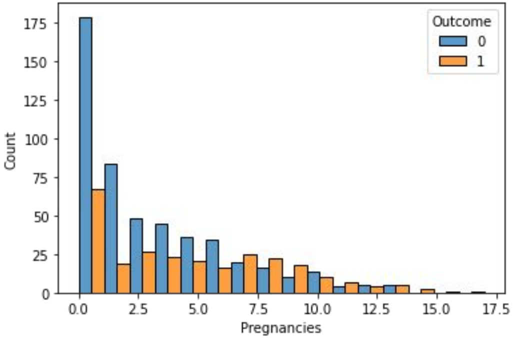
count	500.000000
mean	31.190000
std	11.667655
min	21.000000
25%	23.000000
50%	27.000000
75%	37.000000
max	81.000000



Distribution of Number of Pregnancies and Outcome

Caution: Zeros and Outliers were kept in the data set, but some zeros may be invalid



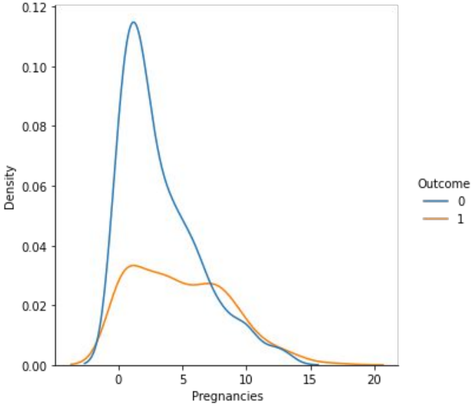


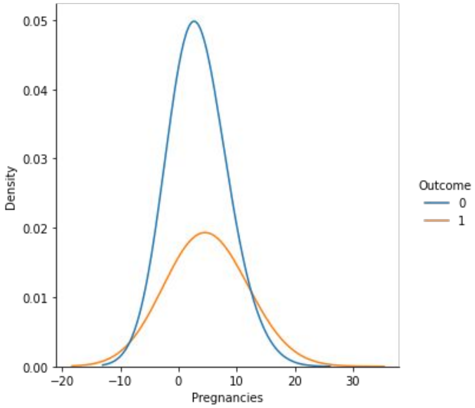
Positive Diabetes:

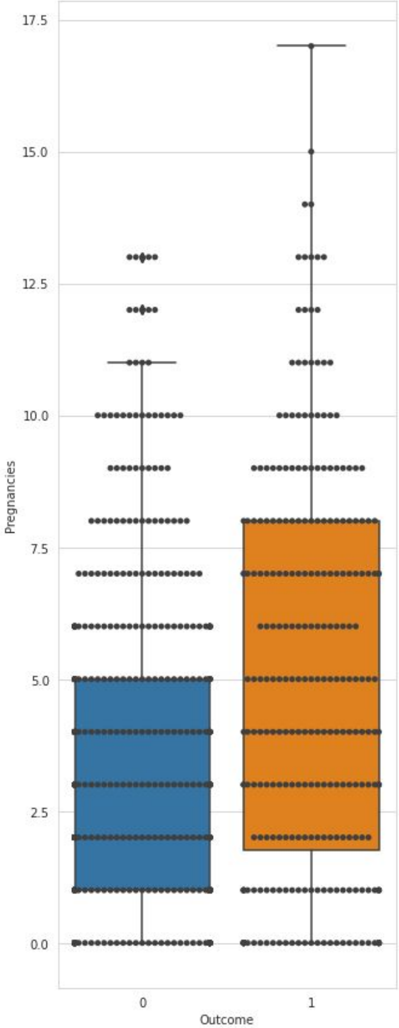
count	266.000000
mean	4.781955
std	3.626607
min	0.000000
25%	1.250000
50%	4.000000
75%	8.000000
max	17.000000

Negative Diabetes:

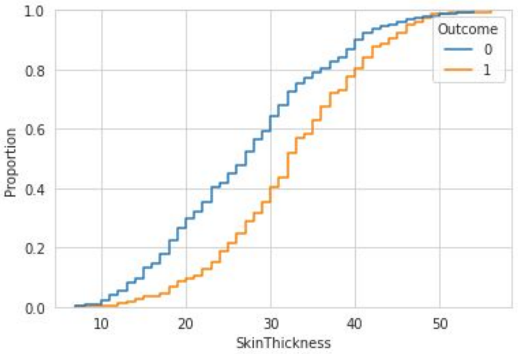
count	500.000000
mean	3.298000
std	3.017185
min	0.000000
25%	1.000000
50%	2.000000
75%	5.000000
max	13.000000

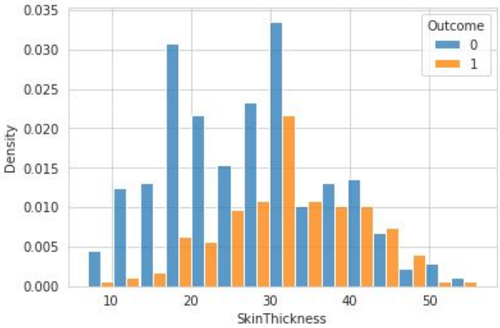


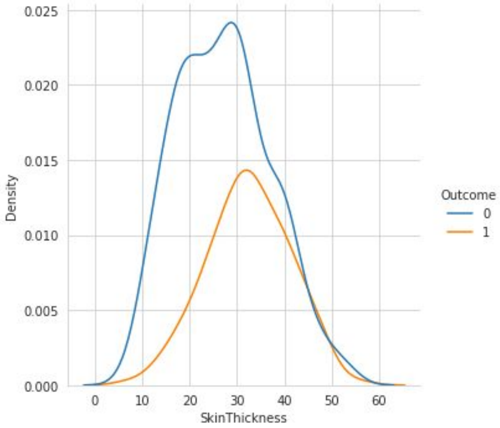


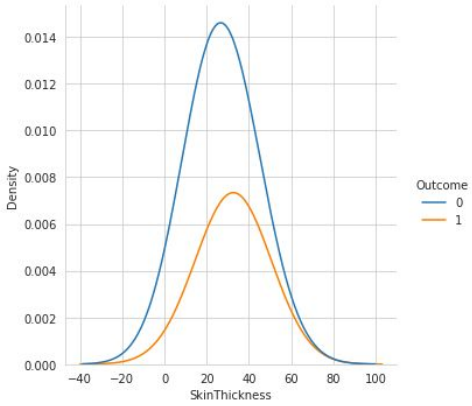


Distribution of Skin Thickness and Outcome



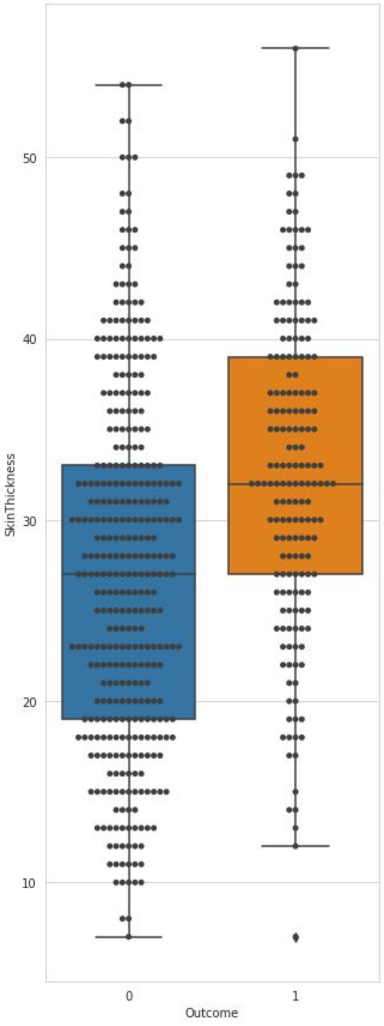






Positive Diabetes:
count 178.000000
mean 32.460674
std 8.824793
min 7.000000
25% 27.000000
50% 32.000000
75% 39.000000
max 56.000000

Negative Diabetes:
count 360.000000
mean 27.144444
std 9.889992
min 7.000000
25% 19.000000
50% 27.000000
75% 33.000000
max 54.000000



Step 2: Inference:

Determining Association between Attributes and Outcome

Step 1: Hypothesis Statement

Let β_1 be the true population linear regression slope between glucose and insulin

H_0: $\beta_1 = 0$ There is no association between glucose and insulin

H_A: $\beta_1 \neq 0$ There is an association between glucose and insulin

Step 2: Assumptions and Conditions

1) Quantitative Data
    a) Glucose: mmol/L - Plasma glucose concentration a 2 hours in an oral glucose tolerance test
    b) Insulin Level: mu U/ml
2) Linearity and Equal Variance
    a) Straight Enough - check original scatterplot and residual scatterplot

Conditions not satisfied! Fanning occurring -- not equal variance.

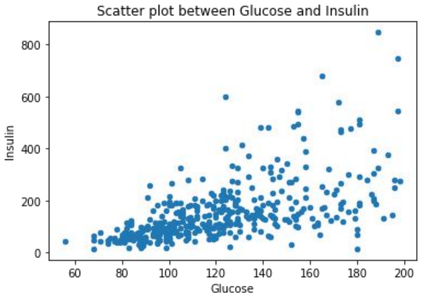
Check: Use an F-test:
Let s_1^2 = variance of glucose
Let s_2^2 = variance of insulin

H_0: The samples have equal variances. $s_1^2 = s_2^2$
H_A: The samples do not have equal variances. $s_1^2 \neq s_2^2$

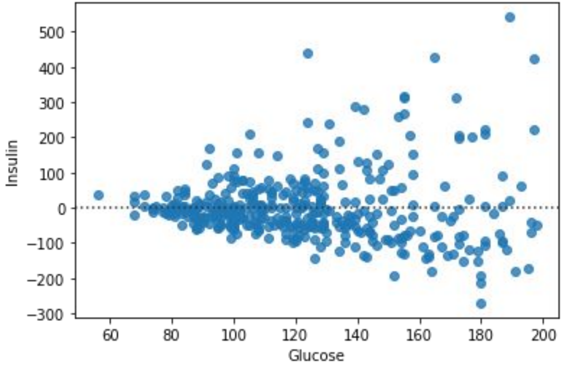
Test statistic:
$F = s_1^2 / s_2^2 = (30.535641)^2 / (118.775855)^2 \sim 0.0661$

Numerator df: $n_1 - 1 = 391$
Denominator df: $n_2 - 1 = 392$
F distribution calculator: $P(F \leq 0.061) \sim 0$

Therefore we have a low P-value and reject the null Hypothesis. There is convincing statistical evidence that the variances are not equal.



Original scatterplot with zeros removed



Residual plot with zeros removed

Hypothesis Statement

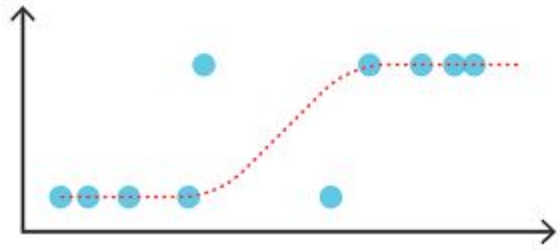
Is there an association between the attributes and diabetes, or only certain ones?

H_0: There is no association between the attributes and diabetes

H_A: There is an association between attributes and diabetes

What is the Logistic Regression Model?

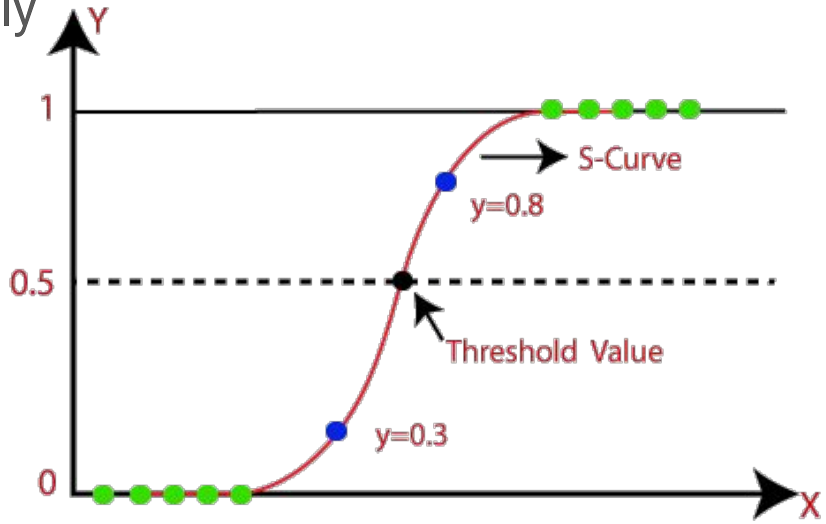
- 3 types: Binomial, Multinomial, Ordinal
- Binomial
  - Predicts two classes (0 or 1)
- Sigmoid used to represent data



Y-variable: either 0 or 1

Logistic Regression for OUR Dataset

- Wanted to see if our data follows a logistic predictive curve
- We knew the data was not linear
- We were interested in finding an equation that could model the distribution of our dataset logistically



Multiple Logistic Regression

Assumption and Conditions:

1) Linearity of natural log of odds - can be proven through high accuracy of regression
2) No Outliers - yes, were removed
3) At least one or more independent variables, but does not need to be all independent
4) No Multicollinearity - yes, no attributes had a simple correlation e.g. glucose vs insulin

p-value

Confidence interval

	coef	std err	z	P>|z|	[0.025	0.975]
Pregnancies	0.1299	0.049	2.655	0.008	0.034	0.226
Glucose	0.0174	0.005	3.765	0.000	0.008	0.026
BloodPressure	-0.0484	0.009	-5.123	0.000	-0.067	-0.030
SkinThickness	0.0284	0.015	1.898	0.058	-0.001	0.058
Insulin	0.0019	0.001	1.598	0.110	-0.000	0.004
BMI	-0.0365	0.022	-1.669	0.095	-0.079	0.006
DiabetesPedigreeFunction	0.4636	0.344	1.347	0.178	-0.211	1.138
Age	0.0005	0.016	0.031	0.976	-0.031	0.032

	Coef.	Std.Err.	z	P>|z|	[0.025	0.975]
Pregnancies	0.1291	0.0374	3.4489	0.0006	0.0557	0.2024
Glucose	0.0215	0.0040	5.4447	0.0000	0.0138	0.0293
BloodPressure	-0.0507	0.0089	-5.6868	0.0000	-0.0682	-0.0332
SkinThickness	0.0299	0.0149	2.0073	0.0447	0.0007	0.0592
BMI	-0.0313	0.0215	-1.4537	0.1460	-0.0734	0.0109

Multiple Logistic Regression

- To identify which variables influence the outcome, we will look at the p-value of each variable. We expect the p-value to be less than 0.05(alpha risk)
    a. Determined an association between Glucose, Blood Pressure and Pregnancies for Diabetes
    b. Accuracy: 77%

Flaws: Not able to obtain complex relationships and relies on linearity of natural log of data, excludes dependent influential variables

Confusion Matrix:

229	33 Type 1 error
59 Type 2 error	71 Power

Final:

	coef	std err	z	P>|z|	[0.025	0.975]
Pregnancies	0.1405	0.037	3.826	0.000	0.069	0.212
Glucose	0.0210	0.004	5.709	0.000	0.014	0.028
BloodPressure	-0.0525	0.007	-7.449	0.000	-0.066	-0.039

	precision	recall	f1-score	support
0	0.80	0.87	0.83	262
1	0.68	0.55	0.61	130
accuracy			0.77	392
macro avg	0.74	0.71	0.72	392
weighted avg	0.76	0.77	0.76	392

Steps 3 - 5: MACHINE LEARNING!!!

Machine Learning Models

- Supervised Learning
- Classification
- Split the data into 70% training, 30% testing
- 4 Different Algorithms
- Standardized Data
- Iterative Optimization Approach
- Data Filtration
- 8 Features, 9th feature → class (0 and 1)
- Feature Selection

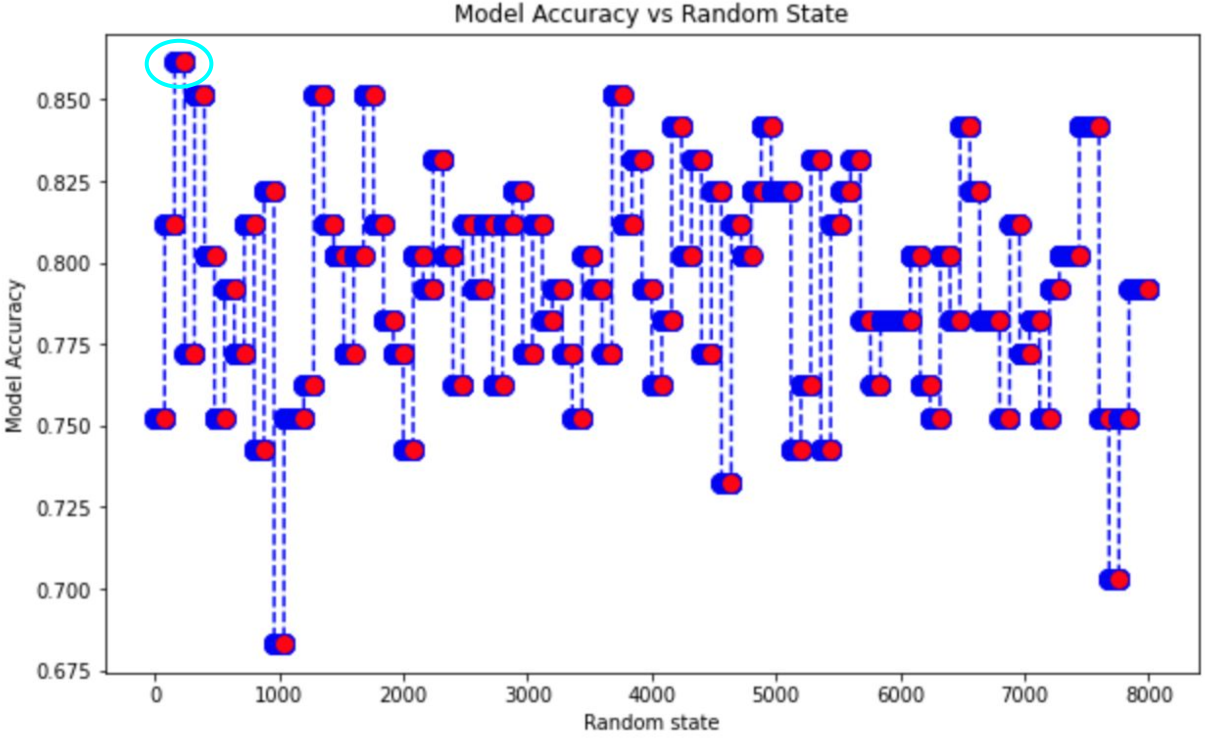
MACHINE LEARNING: Logistic Regression

Logistic Regression for OUR Dataset: FINAL MODEL

- Removed Zeros **and Outliers** from the dataset
- Found out that there was still 336 data entries
- Removed the data: Link to Code Cell
- Trained the model (again) on new data with outliers and zeros removed
  - Used even more repeated hyperparameter tuning
- Final Accuracy: 86.139%
- Graph on next slide
- Yes!! Satisfied!

Logistic Regression Model: FINAL MODEL

```
Maximum accuracy 0.8613861386138614 at random_state 1 = 2 at random_state 2 = 0
```



Step 3: MACHINE LEARNING: Naive Bayesian Model

What is the Naive Bayesian Machine Learning Model?

Bayes Theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- Supervised Learning
- Used for Classification
- Probabilistic Model based on Conditional Probabilities
- Assumption is that the features are independent
  - We proceeded with caution
- Example
  - Weather conditions
  - Series of conditional probabilities

	OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY GOLF
0	Rainy	Hot	High	False	No
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	False	Yes
3	Sunny	Mild	High	False	Yes
4	Sunny	Cool	Normal	False	Yes
5	Sunny	Cool	Normal	True	No
6	Overcast	Cool	Normal	True	Yes
7	Rainy	Mild	High	False	No
8	Rainy	Cool	Normal	False	Yes
9	Sunny	Mild	Normal	False	Yes
10	Rainy	Mild	Normal	True	Yes
11	Overcast	Mild	High	True	Yes
12	Overcast	Hot	Normal	False	Yes
13	Sunny	Mild	High	True	No

Naive Bayesian Machine Learning Model for OUR Dataset

- Wanted to understand if we could predict outcome of 1(Diabetes) or 0(No Diabetes)
  - Using probabilistic relationships!
- The stronger the performance metrics the clearer it was for us to know that there were **underlying connections**

$$X = (x_1, x_2, x_3, ..., x_n)$$

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

Constant denominator

$$P(y|x_1, ..., x_n) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$$

$$y = argmax_y P(y) \prod_{i=1}^{n} P(x_i|y)$$

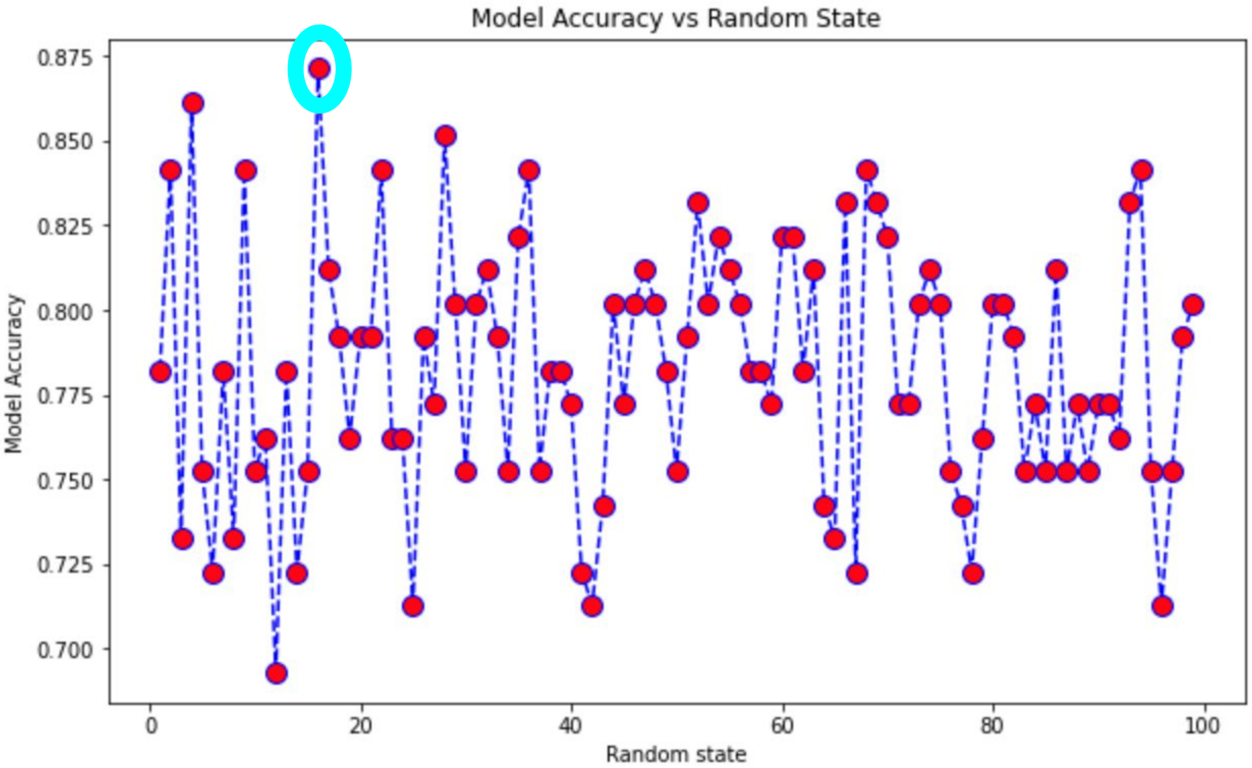
Find maximum class prediction (0 or 1)

Naive Bayesian for OUR Dataset: FINAL MODEL

- Removed Zeros **and Outliers** from the dataset
- Found out that there was still 336 data entries
- Removed the data: Link to Code Cell
- Trained the model (again) on new data with outliers and zeros removed
  - Used even more repeated hyperparameter tuning
- Final Accuracy: 87.129%
- Graph on next slide
- Yes!! Satisfied!

Naive Bayes Model: FINAL MODEL

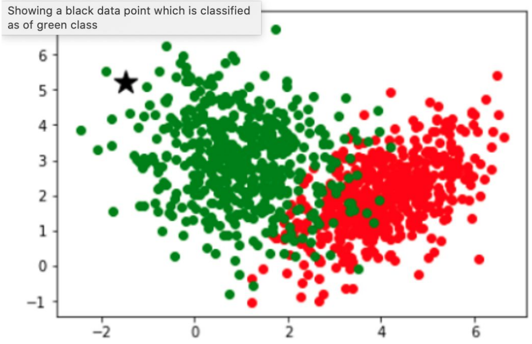
```
Maximum accuracy 0.8712871287128713 at random_state = 16
```



Step 4: MACHINE LEARNING: K-Nearest Neighbors Algorithm

What is the k Nearest Neighbors Machine Learning Model?

- Supervised Machine learning Model
- Classification
- Splits Data based on clusters (0 - No diabetes, 1 - diabetes)
- Takes an incoming point and calculates the k nearest neighbors
  - Used Euclidean Distance
  - Classifies based on the most common class occurrence



k Nearest Neighbors for OUR Dataset

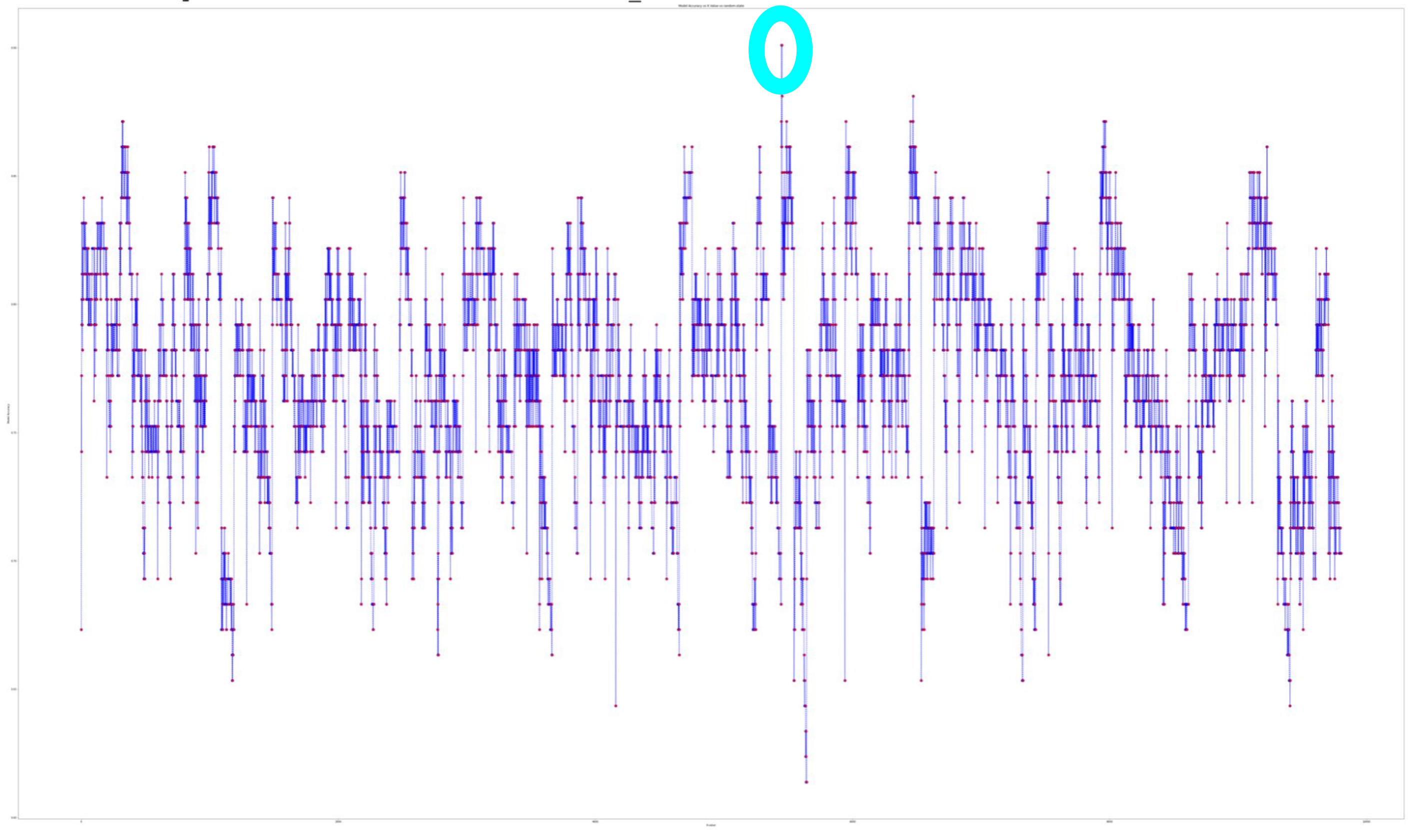
- We wanted to see if a kNN would be a better predictor of our model
- We had more freedom with this algorithm
  - Could tune the hyperparameter of K
- This would also show us that individuals with Diabetes or No Diabetes would have similar conditions
  - “Nearest neighbors”
- After repeated model training sessions
  - Found that it was very sensitive to changes in data/outliers

kNN for OUR Dataset: FINAL MODEL

- Removed Zeros **and Outliers** from the dataset
- Found out that there was still 336 data entries
- Removed the data: Link to Code Cell
- Trained the model (again) on new data with outliers and zeros removed
  - Used even more repeated hyperparameter tuning
- Final Accuracy: 90.099%
- Graph on next slide
- Yes!! Satisfied!

kNN Model: FINAL MODEL

```
Maximum accuracy 0.900990099009901 at K = 4 at random_state = 56
```



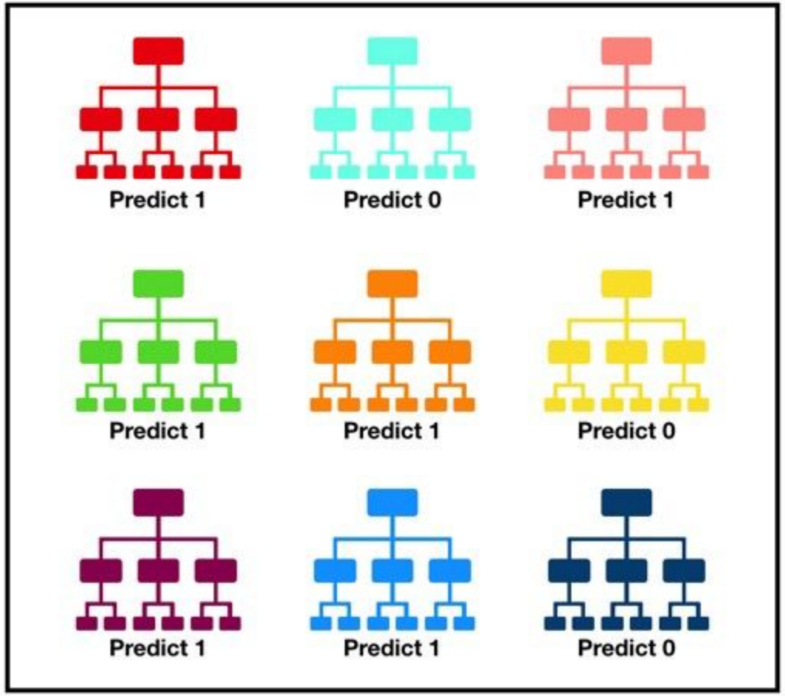
Step 5: MACHINE LEARNING: Random Forest Algorithm with Feature Selection

What is the Random Forest Machine Learning Algorithm?

1. Supervised Learning Model
2. Tree Model (Based on Nodes and Leaves)
3. Random Sampling
4. Steps of Random Forest

How might the Random Forest Model look for our Model?

We are training a lot more than just 6 trees!



Tally: Six 1s and Three 0s

Prediction: 1

Why did we decide to use Random Forests for our Dataset?

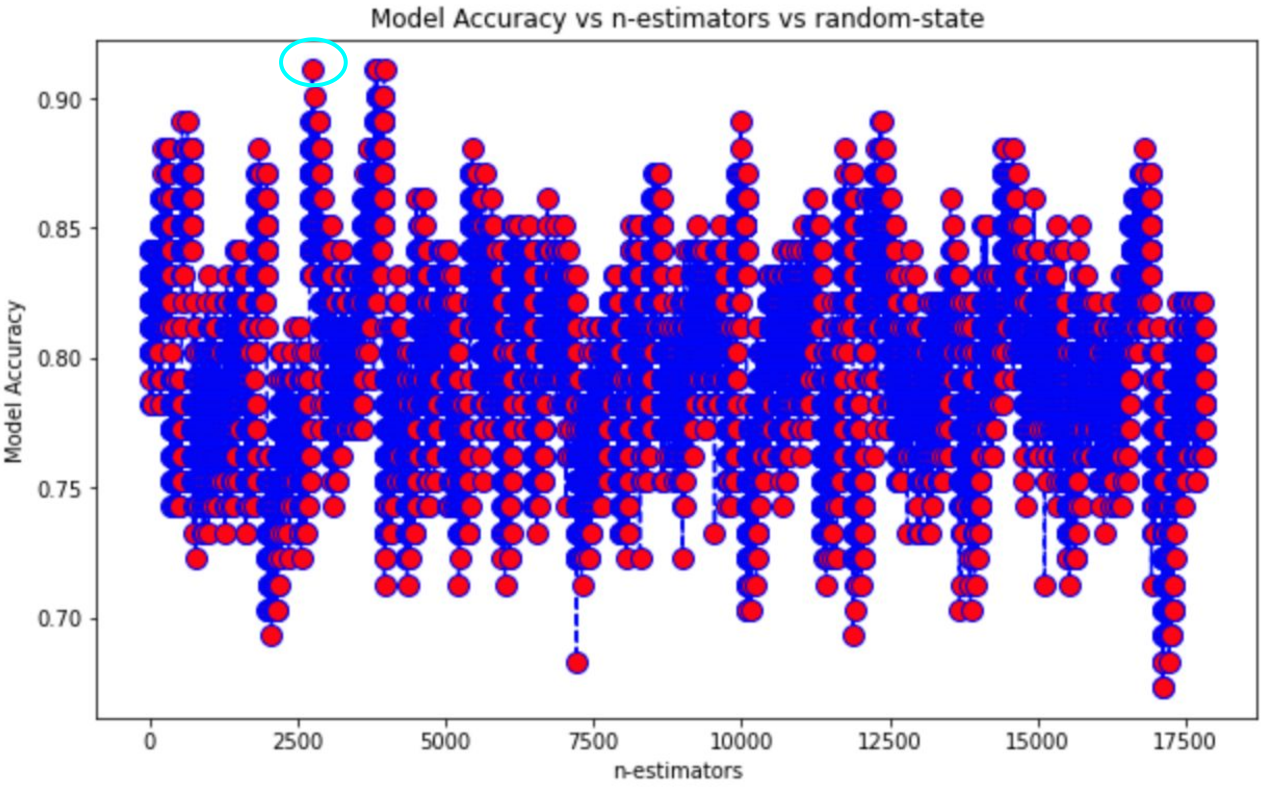
- One of the most common ML Algorithms!
- Predicts really well because it reduces a lot of variability
    - Each tree is independent of each other
        - Random sampling
        - As well as Optimization!!
- Has an inbuilt function known as Feature Selection
    - Allows us to see the most influential features in our dataset!

Random Forests for OUR Dataset: FINAL MODEL

- Removed Zeros **and Outliers** from the dataset
- Found out that there was still 336 data entries
- Removed the data: Link to Code Cell
- Trained the model (again) on new data with outliers and zeros removed
  - Used even more repeated hyperparameter tuning
- Model performed the best with ALL the features
  - No need to do a feature selection for this model
- Final Accuracy: 91.089%
- Graph on next slide
- Yes!! Satisfied!

Random Forests Model: FINAL MODEL

```
Maximum accuracy 0.9108910891089109 at N estimators = 64 at random_state = 16
```



Random Forests - Final Model Code

```
#The loop will break once the highest accuracy is achieved
for i in range (100):
  X_train, X_test, y_train, y_test = train_test_split(X_data_rf, Y_data_rf, test_size=0.3, random_state=16)
  scaler = StandardScaler()
  scaler.fit(X_train)
  X_train = scaler.transform(X_train)
  X_test = scaler.transform(X_test)
  rfc = RandomForestClassifier(n_estimators = 64)
  rfc.fit(X_train, y_train)
  y_pred_combo = rfc.predict(X_test)
  acc = metrics.accuracy_score(y_test, y_pred_combo)
  if (acc > 0.91):
    max_accuracy = acc
    break
print(max_accuracy)
```

```
0.9108910891089109
```

```
[121] print(max_accuracy)
```

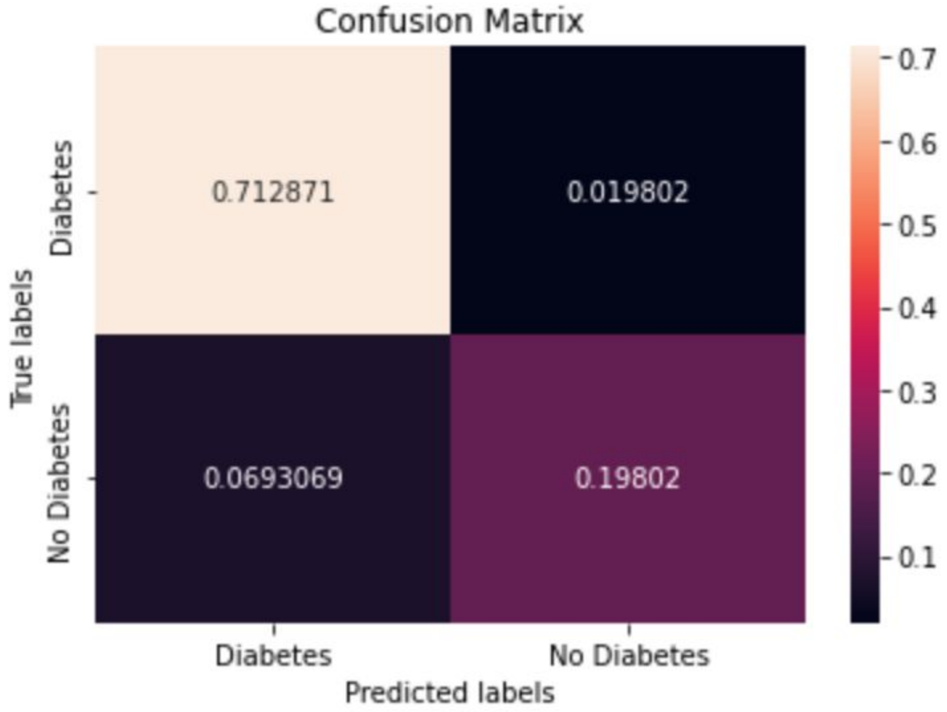
```
0.9108910891089109
```

```
metrics.accuracy_score(y_test, y_pred_combo_final)
```

```
0.9108910891089109
```

Confusion Matrix for Diabetes Prediction: FINAL MODEL

```
[[0.71287129 0.01980198]
 [0.06930693 0.1980198 ]]
```



Conclusion

1. **Random Forests with Tuned Feature Selection**
2. KNN
3. Naive Bayesian
4. Logistic Regression

This is expected given the nature of Random Forests to prevent overfitting while also working well with high-dimensional data.

Future Improvements:

1) Consider varying confounding variables
2) More feature engineering and possible multiple imputations with chained equations as alternative method to account for missing or incorrect data

Thank You!