CS(STAT)5525 : Data Analytics I
Lecture #8

Collegiate Associate Professor
rjafari@vt.edu

Generative v.s Discriminative Model

- Two type of probabilistic classification models:

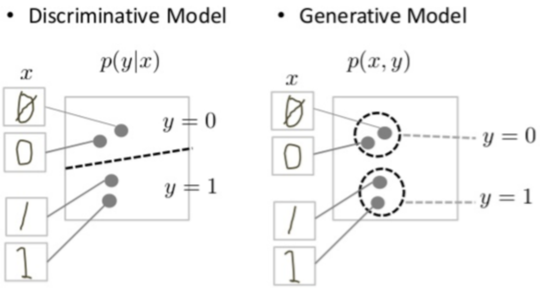
Generative Model

- A Generative model can generate new data instances.
- A statistical model of **joint probability distribution** P(X,Y) on given observable X and target Y.
- Examples : **GANs**, **Naïve Bayes** and **Bayesian Networks**.
- A model of conditional probability of observable X, given a target Y, $P(X|Y = y)$

Discriminative Model

- Discriminative models the **decision boundary between classes**.
- A model of **conditional probability** $P(Y|X = x)$ of the target Y, given an observation x.
- Example : **Logistic Regression**. Capture the process of generating Y given X

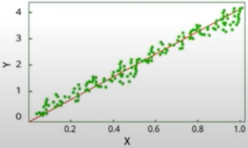
Generative versus Discriminative



Type of Regression

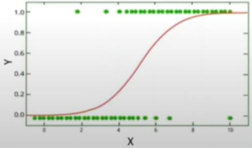
Linear Regression

- When there is a linear relationship between independent and dependent variables.



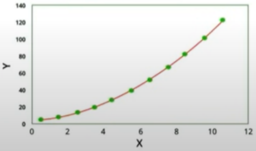
Logistic Regression

- When the dependent variable is categorical (0/ 1, True/ False, Yes/ No, A/B/C) in nature.



Polynomial Regression

- When the power of independent variable is more than 1.



Logistic Regression

- Logistic regression is a classification model that is very easy to implement and performs very well on linearly separable classes.
- It is one of the most widely used algorithms for classification in industry.
- Logistic regression is a linear model for binary classification.
- To explain the idea behind logistic regression as a probabilistic model for binary classification, let define **Odds** in a favor or a particular event:

$$\frac{P(t=1|\mathbf{z})}{P(t=0|\mathbf{z})}$$

- For classifying a test record, it suffices to predict if the odds is > 1 or < 1.

Logistic function

- The goal is to predict the target class t from an input $z = w^T x + b$.

Logistic function

- The goal is to predict the target class t from an input $z = w^T x + b$.
- The probability $P(t = 1|z)$ that input z is classified as class $t = 1$ is represented by the output y of the logistic function computed as

$$y = Pr(t = 1|z) = \sigma(z) = \frac{1}{1 + e^{-z}}$$

where z is the multiple linear regression.

Logistic Regression

- Logistic Regression Model :

$$\frac{P(t=1|\mathbf{x})}{P(t=0|\mathbf{z})} = e^{\mathbf{z}} = e^{\mathbf{w}^T\mathbf{x}+b}$$

where $\mathbf{w}$, b are parameters to be learned during training.

Logistic Regression

- Logistic Regression Model :

$$\frac{P(t=1|\mathbf{x})}{P(t=0|\mathbf{z})} = e^{\mathbf{z}} = e^{\mathbf{w}^T\mathbf{x}+b}$$

  where $\mathbf{w}$, b are parameters to be learned during training.
- It can be proved that :

$$P(t=1|\mathbf{z}) = \frac{1}{1+e^{-z}} = \sigma(z)$$

Logistic Regression

- Logistic Regression Model :

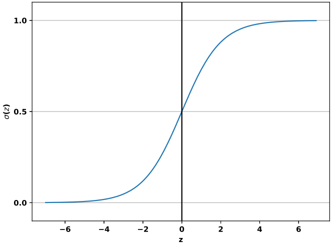
$$\frac{P(t=1|\mathbf{x})}{P(t=0|\mathbf{z})} = e^{\mathbf{z}} = e^{\mathbf{w}^T\mathbf{x}+b}$$

where $\mathbf{w}$, b are parameters to be learned during training.

- It can be proved that :

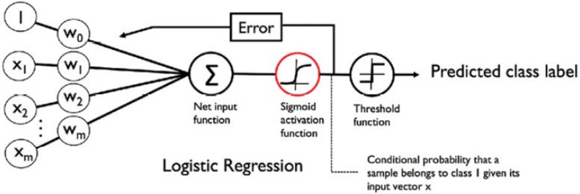
$$P(t=1|\mathbf{z}) = \frac{1}{1+e^{-z}} = \sigma(z)$$

- Sigmoid Function:



Logistic Regression Diagram

- The output of the sigmoid function is then interpreted as the probability of a particular example belonging to class 1, $\sigma(\mathbf{z}) = P(y = 1|\mathbf{x}, \mathbf{w}, b)$
- If the output is 80% for Iris-versicolor flower , therefore, the probability that this flower is an Iris-setosa flower can be calculated as or 20% percent.



Logistic Regression Diagram

- There are many applications where we are not only interested in the predicted class labels, but where the estimation of the class-membership probability is particularly useful.

Logistic Regression Diagram

- There are many applications where we are not only interested in the predicted class labels, but where the estimation of the class-membership probability is particularly useful.
- Logistic regression is used in weather forecasting, for example, not only to predict whether it will rain on a particular day but also to report the chance of rain.

Logistic Regression Diagram

- There are many applications where we are not only interested in the predicted class labels, but where the estimation of the class-membership probability is particularly useful.
- Logistic regression is used in weather forecasting, for example, not only to predict whether it will rain on a particular day but also to report the chance of rain.
- Similarly, logistic regression can be used to predict the chance that a patient has a particular disease given certain symptoms, which is why logistic regression enjoys great popularity in the field of medicine.

Predicted probability

- Let us consider a predictor x and a binary (or Bernoulli) variable y.
- Assuming there exist some relationship between x and y, an ideal model would predict:

$$P(y|x) = \begin{cases} 1: & \text{if y=1} \\ 0: & \text{if y=0} \end{cases}$$

- By using logistic regression, this unknown probability function is modeled as

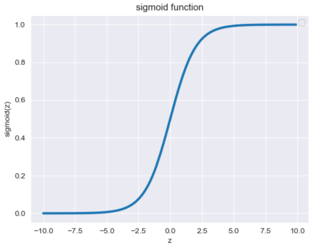
$$\hat{P}(y=1|z) = \sigma(z) = \frac{1}{1+e^{-w^T x}} = \begin{cases} 1: & \text{if } \sigma(z) \geq 0.5 \\ 0: & \text{otherwise} \end{cases}$$

- The predicted probability can then simply be converted into a binary outcome via a threshold function:
- If we look at the preceding plot of the sigmoid function, this is equivalent to the following:

Predicted probability

- If we look at the preceding plot of the sigmoid function, this is equivalent to the following:

$$\hat{y} = \begin{cases} 1: & \text{if } z \geq 0 \\ 0: & \text{otherwise} \end{cases}$$



Learning the weights of the logistic cost function

- Maximize the likelihood of observing training record.

Learning the weights of the logistic cost function

- Maximize the likelihood of observing training record.
- Likelihood of a single record $(\mathbf{x}_i, y_i)$:

$$\begin{aligned}\ell(\mathbf{w}) &= \hat{P}(y_i|\mathbf{x};\mathbf{w}) \\ &= \hat{P}(1|\mathbf{x}_i,\mathbf{w})^y \times \hat{P}(0|\mathbf{x}_i,\mathbf{w})^{1-y}\end{aligned}$$

Learning the weights of the logistic cost function

- Maximize the likelihood of observing training record.
- Likelihood of a single record $(\mathbf{x}_i, y_i)$:

$$
\begin{aligned}
\ell(\mathbf{w}) &= \hat{P}(y_i|\mathbf{x};\mathbf{w}) \\
&= \hat{P}(1|\mathbf{x}_i,\mathbf{w})^y \times \hat{P}(0|\mathbf{x}_i,\mathbf{w})^{1-y}
\end{aligned}
$$

- Likelihood of entire training data :

$$
\begin{aligned}
\ell(\mathbf{w}) &= \prod_{i=1}^{n} \hat{P}(y_i|\mathbf{x}_i,\mathbf{w}) \\
&= \prod_{i=1}^{n} \hat{P}(1|\mathbf{x}_i,\mathbf{w})^{y_i} \times \hat{P}(0|\mathbf{x}_i,\mathbf{w})^{1-y_i}
\end{aligned}
$$

Logistic Regression Model

- The goal is to find $\mathbf{w}$ that maximizes likelihood function. In practice, we minimizes negative of the (natural) log of likelihood function (**Cross-entropy Function**) (Let $t_i \rightarrow$ target and $y_i = \sigma(z_i)$)

$$
\begin{aligned}
J(\mathbf{w}) &= -\sum_{i=1}^{n} [y_i \log(y_i) + (1 - y_i) \log(1 - y_i)] \\
&= -\sum_{i=1}^{n} [y_i \log(\sigma(z_i)) + (1 - y_i) \log(1 - \sigma(z_i))]
\end{aligned}
$$

Logistic Regression Model

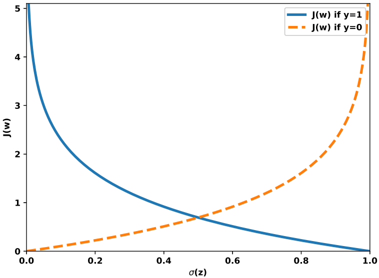
- Learn $\mathbf{w}^*$ that minimize cross-entropy.
- For a single training example, we can see that the first term is zero if $y = 0$ and the second term becomes zero if $y = 1$

$$J(\mathbf{w}) \quad = -\sum_{i=1}^{n} \left[y_i \log(\hat{P}(1|\mathbf{x}_i, \mathbf{w})) + (1 - y_i) \log(\hat{P}(0|\mathbf{x}_i, \mathbf{w})) \right]$$

$$J(\mathbf{w}) = \begin{cases} -\log(P(1\hat{|}\mathbf{x}_i, \mathrm{w}) & \text{if } y_i = 1 \\ -\log(1 - P(1\hat{|}\mathbf{x}_i, \mathrm{w}) & \text{if } y_i = 0 \end{cases}$$

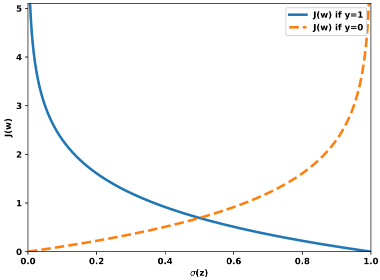
Logistic Regression Model

- It can be observed that the cost function $\to 0$ (continuous line) if we correctly predict that an example belongs to class 1.



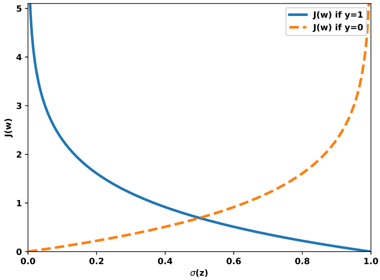
Logistic Regression Model

- It can be observed that the cost function $\rightarrow 0$ (continuous line) if we correctly predict that an example belongs to class 1.
- Similarly, we can see on the y-axis that the cost also approaches 0 if we correctly predict $y = 0$ (dashed line).



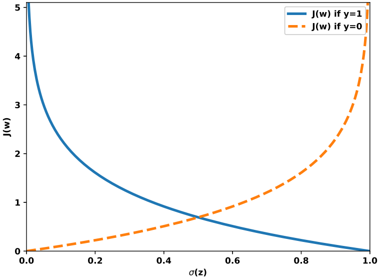
Logistic Regression Model

- It can be observed that the cost function $\rightarrow 0$ (continuous line) if we correctly predict that an example belongs to class 1.
- Similarly, we can see on the y-axis that the cost also approaches 0 if we correctly predict $y = 0$ (dashed line).
- However, if the prediction is wrong, the cost goes toward infinity.



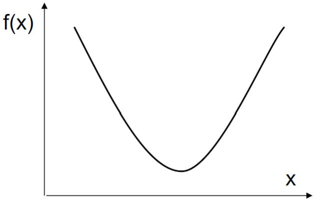
Logistic Regression Model

- It can be observed that the cost function $\rightarrow 0$ (continuous line) if we correctly predict that an example belongs to class 1.
- Similarly, we can see on the y-axis that the cost also approaches 0 if we correctly predict $y = 0$ (dashed line).
- However, if the prediction is wrong, the cost goes toward infinity.
- The main point is that we penalize wrong predictions with an increasingly larger cost.

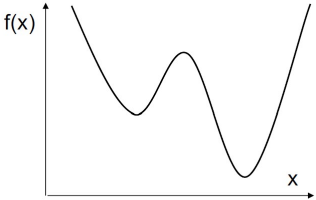


Learning Logistic Model as Convex Optimization

- The minimization of the cross-entropy function is a Convex optimization problem.
- A convex optimization problem:
  - Every local minima is a global minima.
  - Can be solved using standard optimization techniques such as Gradient Descend or Newton's Method.
  - Start with initial solution of model parameters.
  - Update parameters in direction of steepest descend.
  - Converge when gradient is 0 (local minima)



Convex

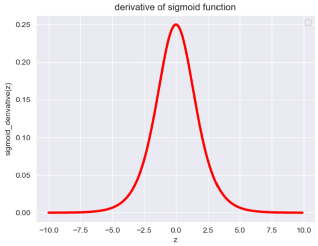


Non-convex

Derivative of Logistic function

- For the gradient descent optimization technique to minimize the logistic function, the derivative with respect to input z can be calculated as:

$$\frac{\partial y}{\partial z} = \frac{\partial \sigma(z)}{\partial z} = \sigma(z)(1 - \sigma(z))$$



Gradient Descent Learning Algorithm for Logistic Regression

- Using calculus, optimum weights can be found by calculating the partial derivative of the log-likelihood function with respect to the jth weight:

$$\frac{\partial}{\partial_j} J(\mathbf{w}) = \left(y \frac{1}{\sigma(z)} - (1-y) \frac{1}{1-\sigma(\mathbf{z})}\right) \frac{\partial}{\partial w_j} \sigma(\mathbf{z})$$

- To update all weights simultaneously, we can write the general update rule as follows:

$$\mathbf{w} := \mathbf{w} + \Delta \mathbf{w}$$

where $\Delta \mathbf{w} = \eta \nabla J(\mathbf{w})$

- Maximizing log-likelihood is equal to minimizing the cost function $J(\mathbf{w})$. Hence:

$$\mathbf{w} := \mathbf{w} - \Delta \mathbf{w}$$

Characteristics of Logistic Regression

- Does not make any assumption about conditional probabilities and directly computes the posterior probability $P(Y|X)$

Characteristics of Logistic Regression

- Does not make any assumption about conditional probabilities and directly computes the posterior probability $P(Y|X)$
- Can handle interacting variables.

Characteristics of Logistic Regression

- Does not make any assumption about conditional probabilities and directly computes the posterior probability $P(Y|X)$
- Can handle interacting variables.
- Can handle irrelevant and redundant attributes, as long as we can avoid overfitting.

Characteristics of Logistic Regression

- Does not make any assumption about conditional probabilities and directly computes the posterior probability $P(Y|X)$
- Can handle interacting variables.
- Can handle irrelevant and redundant attributes, as long as we can avoid overfitting.
- Robust to high dimensional attributes as it does not involve computing density or distances of points.

Characteristics of Logistic Regression

- Does not make any assumption about conditional probabilities and directly computes the posterior probability $P(Y|X)$
- Can handle interacting variables.
- Can handle irrelevant and redundant attributes, as long as we can avoid overfitting.
- Robust to high dimensional attributes as it does not involve computing density or distances of points.
- Cannot handle missing values

Characteristics of Logistic Regression

- Does not make any assumption about conditional probabilities and directly computes the posterior probability $P(Y|X)$
- Can handle interacting variables.
- Can handle irrelevant and redundant attributes, as long as we can avoid overfitting.
- Robust to high dimensional attributes as it does not involve computing density or distances of points.
- Cannot handle missing values
- Can only learn linear decision boundaries.

ROC curve

- A receiver operating characteristic curve (ROC) curve is a graphical plot that illustrates the performance of a classification model at all classification thresholds.
- This curve plots two parameters: True Positive Rate and False Positive Rate
- **True Positive Rate (TPR)** is a synonym for **recall**:

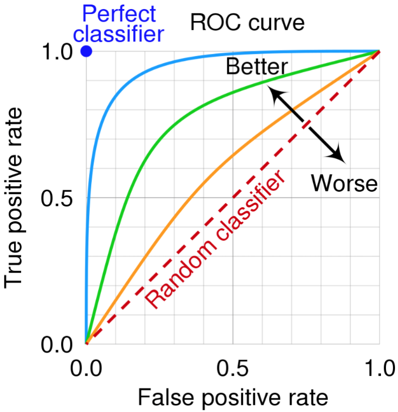
$$TPR = \frac{TP}{TP + FN}$$

- **False Positive Rate (FPR)** is defined as:

$$FPR = \frac{FP}{FP + TN}$$

ROC curve

- An ROC curve plots TPR versus FPR at different classification thresholds.
- Lowering the classification threshold classifies more items as positive, thus increasing both False Positive and True Positive.



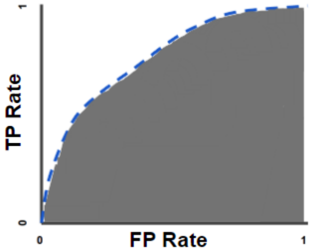
ROC curve example

- The result of model B is on the line. Random guess ACC = 50%.
- Model A performs better than B and C.
- Model C' performs the best.

A			B			C			C′		
TP=63	FN=37	100	TP=77	FN=23	100	TP=24	FN=76	100	TP=76	FN=24	100
FP=28	TN=72	100	FP=77	TN=23	100	FP=88	TN=12	100	FP=12	TN=88	100
91	109	200	154	46	200	112	88	200	88	112	200
TPR = 0.63			TPR = 0.77			TPR = 0.24			TPR = 0.76		
FPR = 0.28			FPR = 0.77			FPR = 0.88			FPR = 0.12		
PPV = 0.69			PPV = 0.50			PPV = 0.21			PPV = 0.86		
F1 = 0.66			F1 = 0.61			F1 = 0.23			F1 = 0.81		
ACC = 0.68			ACC = 0.50			ACC = 0.18			ACC = 0.82		

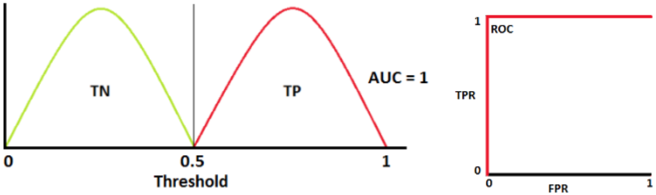
AUC curve

- To compute the points in an ROC curve., we could evaluate a logistic regression model many times with different classification thresholds.
- But this would be inefficient. There is an efficient sorting-based algorithm than can provide this information, called Area Under the ROC Curve (AUC).
- **AUC**measures the entire two-dimensional area underneath the entire ROC curve (think integral calculus) from (0,0) to (1,1)



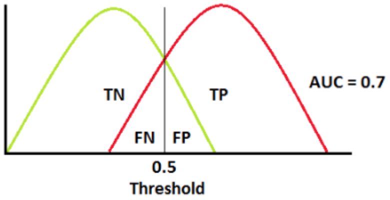
How to speculate about the performance of the model?

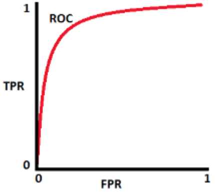
- An excellent model has AUC near to the 1 which means it has a good measure of separability.
- A poor model has an AUC near 0 which means it has the worst measure of separability. It is predicting 0s as 1s and 1s as 0s.
- And when AUC is 0.5, it means the model has no class separation capacity whatsoever.
- In an situation:



Understating of AUC

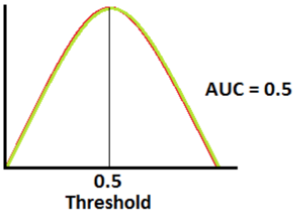
- When two distributions overlap, we introduce type 1 and type 2 errors.
- Depending upon the threshold, we can minimize or maximize them.
- When AUC is 0.7, it means there is a 70% chance that the model will be able to distinguish between positive class and negative class.

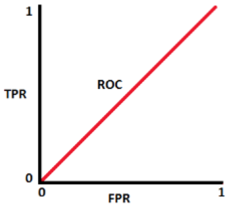




Understating of AUC

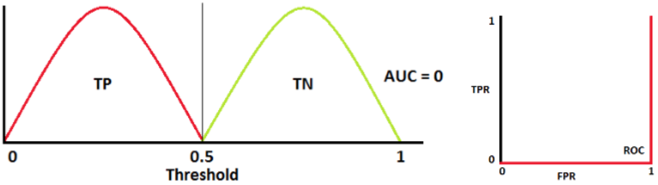
- This is the worst situation.
- When AUC is approximately 0.5, the model has no discrimination capacity to distinguish between positive class and negative class.





Understating of AUC

- When AUC is approximately 0, the model is actually reciprocating the classes.
- It means the model is predicting a negative class as a positive class and vice versa.



5 questions in data analytics