In this presentation, we will talk about optimization algorithms that are used to minimize loss functions in machine learning models.

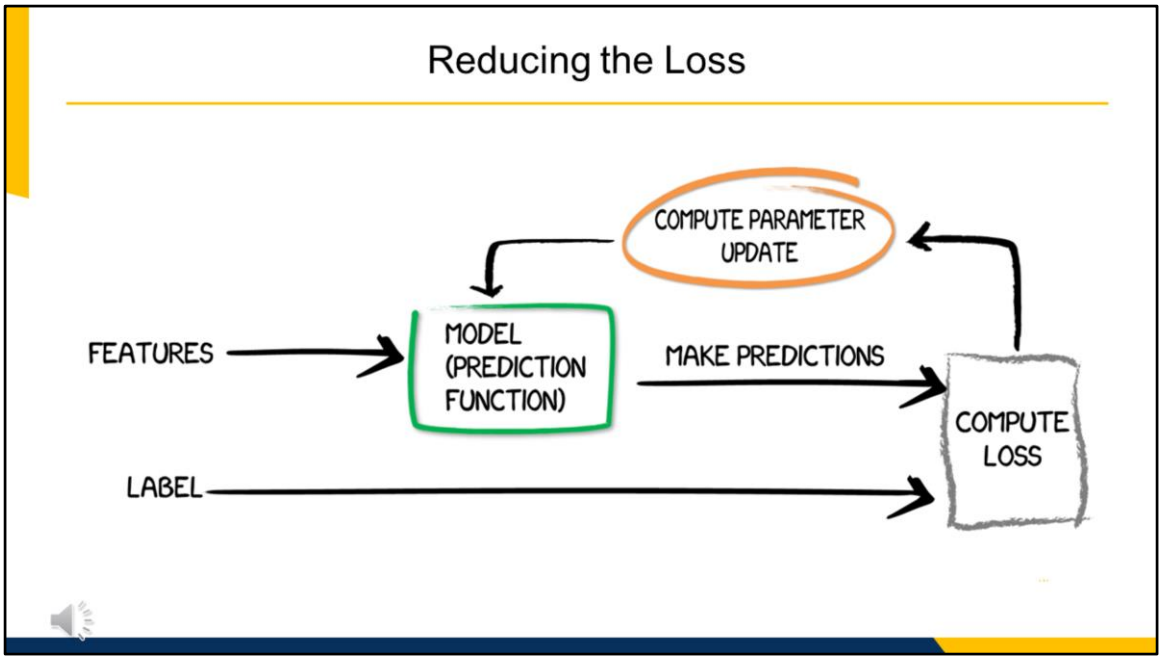
Reducing the Loss

- We know that model training involves choosing the model parameters so that to reduce the loss. But how?
- A good analogy is the "Hot and Cold" kid's game for finding a hidden object
- In this game, the "hidden object" is the best possible model that leads to minimum loss.
- We will start with a wild guess for model parameters, make predictions, and wait for the model to tell us what the loss is.
- Then, we will update the model's parameters and try again, hopefully doing better this time (i.e., we get warmer)

KENT STATE UNIVERSITY

So far we understood that training a model involves choosing model parameter that leads to minimizing some loss or cost function. We also discussed a number of common loss metrics for classification and regression models. In this presentation, we want to learn more about the process of optimization. In some scenarios such as minimizing the root square of errors for a linear regression model, we can indeed solve the optimization problem analytically. This involves finding the derivative of the cost function and solving for zero. However, in most cases, the optimization is heuristic meaning that we need to rely on various search algorithm that can help us find the minimum point of the cost function.

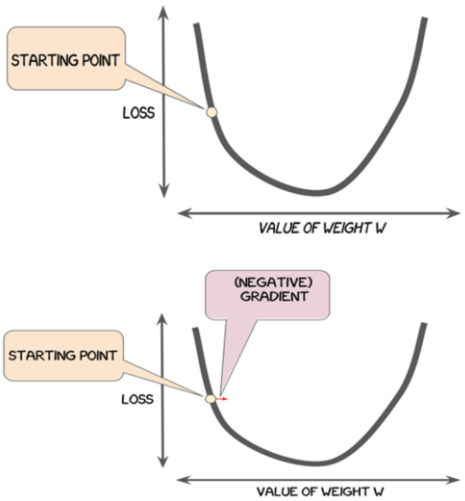
A good analogy is the "Hot and Cold" kid's game for finding a hidden object. In this game, the "hidden object" is the best possible model that leads to minimum loss. We will start with a wild guess for model parameters, make predictions, and wait for the model to tell us what the loss is.
Then, we will update the model's parameters and try again, hopefully doing better this time. The process is repeated until the optimal point is found.



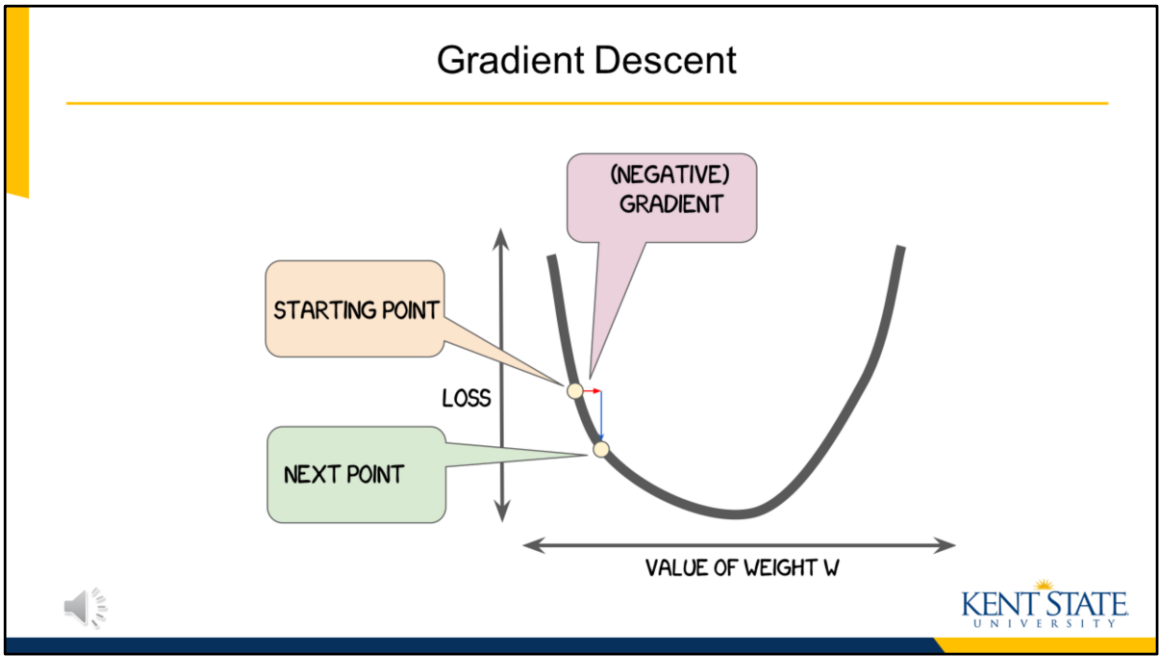
Here is a high level overview of the optimization process. As shown, input variables or features are fed into a model. The model choose some parameters and produce predictions, the predictions are then compared against the actual labels and the loss values re calculated. We will then use this information to update the model's parameters. The process continues until we hopefully find the parameter set for the model that corresponds to the minimum value of the loss.

Gradient Descent

- Gradient descent is an optimization algorithm used to minimize some function by iteratively moving in the direction of steepest descent as defined by the negative of the gradient.
- Lets assume we want to find the weight of a linear regression model.
- The first stage in gradient descent is to pick a starting value for the weight.
- The gradient descent algorithm then calculates the gradient (slope) of the loss curve at the starting point.

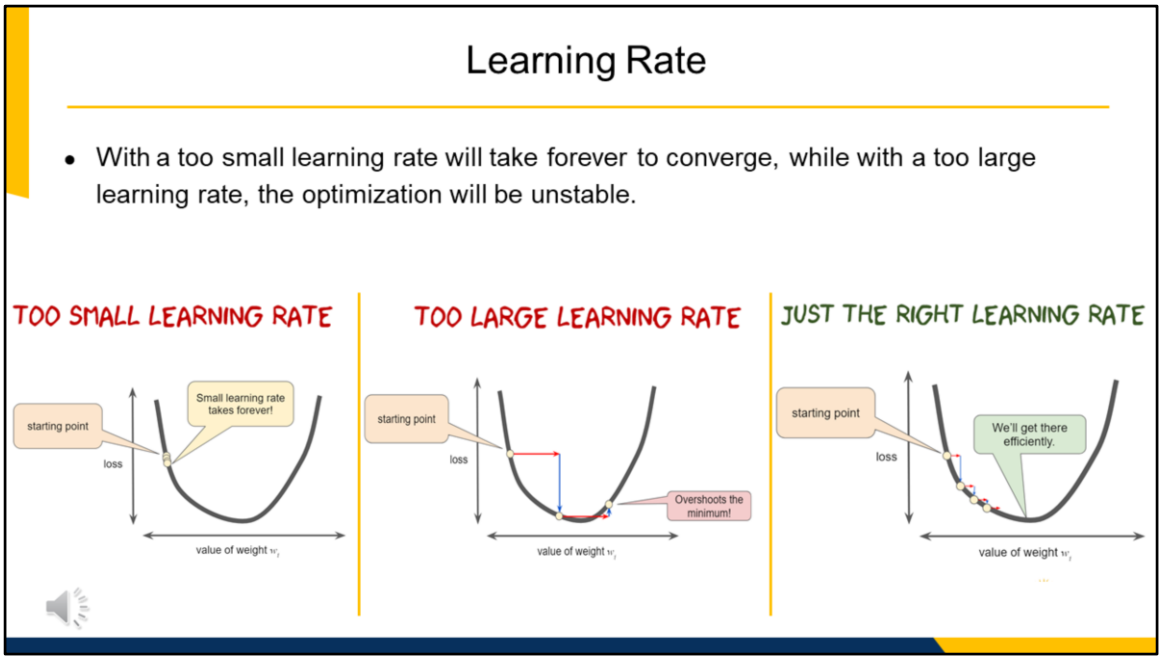


Now let us introduce Gradient Descent which is known to be one of the most common optimization algorithms. As the name implies, gradient descent is an optimization algorithm used to minimize some function by iteratively moving in the direction of steepest descent as defined by the negative of the gradient. Suppose we want to find the weights or the coefficients of a linear regression model.
The first stage in gradient descent is to pick a starting value for the weights.
The gradient descent algorithm then calculates the gradient (slope) of the loss curve at the starting point.



Once negative gradient direction is determined, the model's parameters are changed in that direction. In this example, the linear regression weight or coefficient, W is increased since this is the negative of the gradient of the loss function.

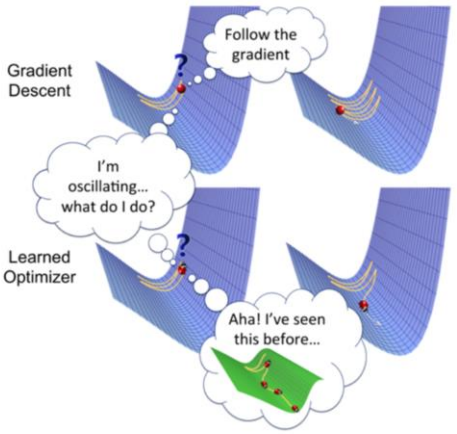
The commonly used analogy to describe Gradient Descent is hiking down a hill from an initial starting point, while choosing a direction to advance using small steps along the way toward a minimum point. The gradient descent process uses the derivatives of the cost function to follow the function downhill to a minimum.



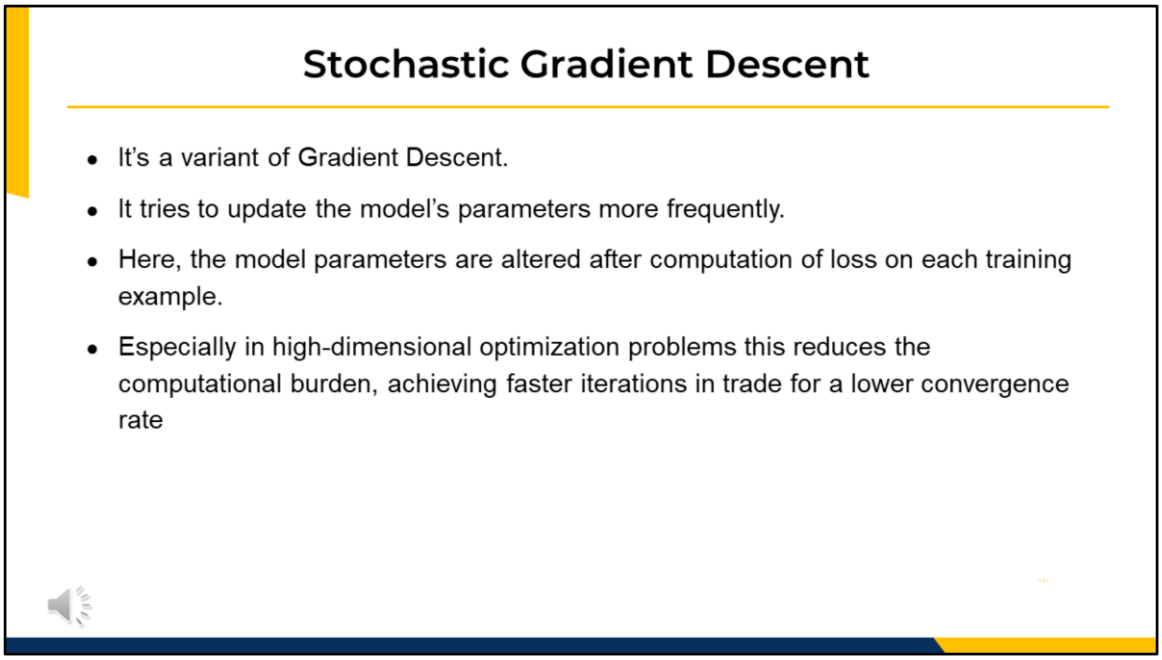
The gradient tells us in which direction we need to go. In this example we know that we need to increase the linear regression weight or coefficient. But by how much? Well this is determined based on another parameter called the learning rate. The lower the value of the earning rate, the smaller will be our step and the slower we travel along the downward slope. While this might be a good idea in terms of making sure that we do not miss the minimum point it could also mean that we'll be taking a long time to converge. On the other hand, if the learning rate is too large, we may overshoot the minimum and miss it.

Oscillating Around the Optimum

- Oscillating around the optimal value may lead the gradient decent algorithmm not to converge.
- The native algorithm can be enhanced by providing the opportunity to detect oscillations around the optimal value.



As we discussed, if the learning rate is too high, we may constantly oscillate around the minimum point. As such, the native algorithm can be enhanced by providing the opportunity to detect oscillations around the optimal value and change the learning rate accordingly to avoid such oscillations. It is also a common practice to start with a high learning rate to get closer to the optimal region quickly and then to reduce the learning rate to allow the algorithm to make small changes to find the minimum.



Stochastic Gradient Descent or SGD for short is a variant of Gradient Descent algorithm. The algorithm tries to update the model's parameters more frequently. More specifically, the model parameters are altered after computation of loss on each training example. This will reduce the overall computational complexity of the search process which will be highly desirable especially in high dimensional optimization problem where we have a large number of input variables.

Even though SGD has been around in the machine learning community for a long time, it has received a considerable amount of attention just recently in the context of large-scale learning.
SGD has been successfully applied to large-scale and sparse machine learning problems often encountered in text classification and natural language processing problems.


KENT
STATE
UNIVERSITY
Thank You.

Reducing Loss: Classification Models

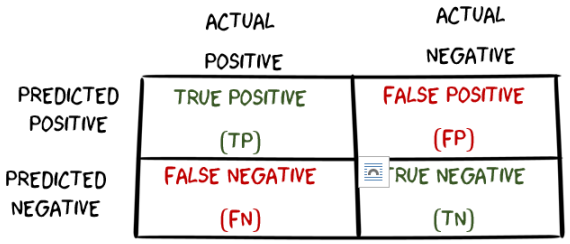
KENT STATE UNIVERSITY

In this presentation, we will discuss various loss functions and performance metrics for classification models.

Loss Metrics: Classification Tasks

- Let's consider a binary classifier that seeks to predict an outcome that is either positive (e.g., loan default) or negative (e.g., non-default)

- There will be four possibilities for each predictions



KENT STATE UNIVERSITY

Let's consider a binary classifier that seeks to predict an outcome that is either positive or negative. A confusion matrix is a table with 4 different combinations of predicted and actual values. The confusion matrix is beneficial in providing insights about the performance of a classifier. Let's discuss all 4 combinations of the confusion matrix. First, we have true positives, which are the observations that were predicted as positive, and indeed they were positive. Similarly, true negatives are observations that were predicted as negative, and in fact, they were negative. False positives, on the other hand, are negative observations that are predicted as positive. False positives are also known as Type I errors. Finally, false negatives or Type II errors are positive observations that are predicted as negative.

Accuracy

- Accuracy is the proportion of true predictions, either true positive or true negative

$$\text{ACCURACY} = \frac{TP+TN}{TP+TN+FP+FN}$$

	ACTUAL POSITIVE	ACTUAL NEGATIVE
PREDICTED POSITIVE	500 (TP)	50 (FP)
PREDICTED NEGATIVE	20 (FN)	850 (TN)

$$\text{ACCURACY} = \frac{500+850}{500+850+50+20} = 0.9507$$

KENT STATE UNIVERSITY

Once you have a confusion matrix, it is easy to calculate different performance measures. Accuracy is a common classification performance metric representing the number of correct predictions to the total number of input samples. More specifically, accuracy can be calculated as the sum of true positives and true negatives divided by all the total number of observations. Let's consider an example where we have 500 true positives, 50 false positives, 20 false negatives, and 850 true negatives. In this example, the classification accuracy is 500 plus 850 divided by the total number of cases, which is calculated as 0.95 or 95% approximately.

Accuracy: Shortcomings

- The cost of False Positives and False Negatives may not be equal. Accuracy, does not account for this.
- Example: In case of the Business Bank loan approval scenario, the cost of approving a loan for an applicant who default (false negative) is substantially higher that rejecting a creditworthy applicant (false positive), assuming that the positive class is default.
- Accuracy is not a good measure when there are class imbalances.
- Example: If only 5% of applicants default, the accuracy of a dumb model that classify all applicants as non-default will be 0.95!

KENT STATE UNIVERSITY

While accuracy is a common metric, it does not differentiate between false positives and false negatives. It simply treats both error types the same. This may not be ideal since, in many business applications, the cost of the two error types are different. For example, considering the earlier loan approval scenario, the cost of approving a loan for an applicant who default is substantially higher than rejecting a creditworthy applicant.

In addition to this, classification accuracy alone doesn't tell the full story when you're working with a class-imbalanced data set, where there is a significant disparity between the number of positive and negative labels. For example, if only 5% of applicants default, the accuracy of a dumb model that classifies all applicants as non-default will be 0.95 or 95%, which may seem high for such a useless model. These types of problems are examples of the fairly common case in data science when accuracy is not a good measure for assessing model performance.

Precision & Recall

- Precision is the fraction of correct classified instances among the ones classified as positive

$$\text{PRECISION} = \frac{TP}{TP+FP}$$

- Recall (also known as Sensitivity or True Positive Rate) is a metric that quantifies the number of correct positive predictions made out of all actual positive cases.

$$\text{RECALL} = \frac{TP}{TP+FN}$$

EXAMPLE:

	ACTUAL POSITIVE	ACTUAL NEGATIVE
PREDICTED POSITIVE	500 (TP)	50 (FP)
PREDICTED NEGATIVE	20 (FN)	850 (TN)

$$\text{PRECISION} = \frac{500}{500+50} = 0.909$$

$$\text{RECALL} = \frac{500}{500+20} = 0.961$$

KENT STATE UNIVERSITY

To address the shortcomings of the classification accuracy as the single metric, we can use precision and recall to get a better understanding of the model performance. Precision attempts to answer the following question: what proportion of positive predictions was actually correct? As such Precision is defined as the ratio of true positives to the sum of true positives and false positives. Let's consider the previous confusion matrix. In that example, we had 500 true positives and 50 false positives. As such, the precision is 500 divided by 500 plus 50, which is 0.909. In a practical sense, precision tells you how much you can trust your classifier when it tells you an instance belongs to the positive class. A high precision value means there were very few false positives, and the classifier is very strict in the criteria for classifying something as positive.

Now let's take a look at recall.

The recall is defined as the ratio between all the instances that were correctly classified in the positive class against the total number of actual members of the positive class. In other words, it tells you how many of the total numbers of positive instances were correctly classified. As such, recall is calculated as the ratio of true positives divided by the sum of true positives and false negatives.

Let's make another brief analysis of the equation to understand it better.

True Positives are all the classes that were correctly classified as positive. Now we

need to divide it by the total number of actual members of the positive class. If you look at the previous confusion matrix, you can see that the total number of members of the positive class is given by the sum of the true positives and false negatives that is 500 plus 20. Therefore the recall is 500 divided by 520 which is 0.961. In a practical sense, precision tells you how much you can trust your classifier to find all the members of the positive class. A high recall value means there were very few false negatives and that the classifier is more permissive in the criteria for classifying something as positive.

To fully evaluate the effectiveness of a model, you must examine both precision and recall. Unfortunately, precision and recall are often in tension. In other words, having very high values of precision and recall is very difficult in practice, and often you need to choose which one is more important for your application. Usually, increasing the value of precision decreases the value of recall and vice-versa.

One easy mental model for understanding this tradeoff is to imagine how *strict* the classifier is. If the classifier is *very strict* in its criteria to put an instance in the positive class, you can expect a high value in precision: it will filter out a lot of false positives. At the same time, some members of the positives class will be classified as negatives, which results in false negatives, that will reduce the recall.

If the classifier is *very permissive* it will find as many instances in the positive class as possible, but this also means that you run the risk of misclassifying instances of the negative class as positive. This will give you a good recall since you correctly classified almost every positive instance, but it will reduce the precision since there will be more members of the negative class classified as positive.

In practice, you need to understand which metric is more important for your problem and optimize your model accordingly. As a rule of thumb, if missing positive instances is unacceptable, you want to have a high recall. If what you want is more confident in your true positives, then you should optimize for precision. For example, a cancer diagnosis is a problem where false positives are more acceptable than false negatives. It's ok if you misclassify healthy people as members of the positive class, since it may only require a follow-up test. Missing a person who needs treatment, on the other hand, is something you don't want. In this type of problem you want very high recall values: find as many members of the positive class as possible.

The opposite scenario is spam classification, where false negatives are much more tolerable than false positives. It doesn't matter if you occasionally find a spam email in your inbox, but having good emails classified as spam can be problematic. This scenario favors precision over recall.

Now that you have learned about precision and recall, let us re-examine the previous example where we had a dumb model that was classifying all cases as negatives. In that case, the true positive was zero since every observation was classified as negative, and therefore recall would be zero. In this example, precision would be undefined since both the nominator and denominator would be zero.

Specificity & F-Score

- Specificity (or True Negative Rate), is the proportion of actual negatives that are correctly identified as such

$$\text{SPECIFICITY} = \frac{TN}{TN+FP}$$

- The F-score is the harmonic mean of the precision and recall

$$F1 = 2 \times \frac{\text{RECALL} \times \text{PRECISION}}{\text{RECALL} + \text{PRECISION}}$$

EXAMPLE:

	ACTUAL POSITIVE	ACTUAL NEGATIVE
PREDICTED POSITIVE	500 (TP)	50 (FP)
PREDICTED NEGATIVE	20 (FN)	850 (TN)

$$\text{PRECISION} = \frac{500}{500+50} = 0.909$$

$$\text{RECALL} = \frac{500}{500+20} = 0.961$$

$$\text{SPECIFICITY} = \frac{850}{850+50} = 0.944$$

$$F1 = 2 \times \frac{0.961 \times 0.909}{0.961 + 0.909} = 0.934$$

Specificity is calculated as the number of correct negative predictions (TN) divided by the total number of negatives (N). It is also called true negative rate.
In the context of the cancer diagnosis classification problem specify answers the following question: Of all the people who are healthy, how many of those did we correctly predict? In the previous confusion matrix example, we had 850 true negative cases and 50 false positives so the specificity is calculated as 0.944.
Finally, we have another metric that is called the F1 score, which is a function of Precision and Recall. More specifically, the F-score is the harmonic mean of precision and recall. Put another way, the F1 score conveys the balance between precision and recall. If we are looking to select a model based on a balance between precision and recall, the F1 measure can be used as a single metric that combines Precision and Recall. As you can see here, the F1 score is 0.934 in our example.

Classification Threshold

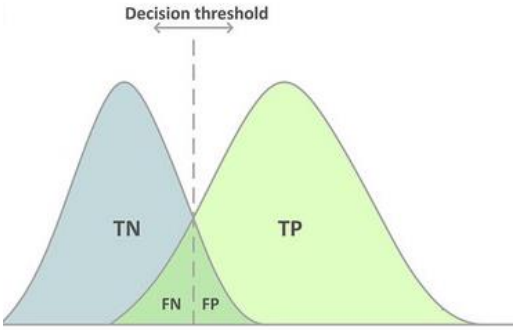
- Most classification models generate probability values as their output (e.g., probability of loan default for an applicant)
- A threshold (i.e., cut-off) is then applied to convert probabilities to binary classes.
- For example, if the threshold is 0.5, Tom's application will be approved if the probability of default for his loan application (as determined by the model) is smaller than 0.5.
- However, since the costs of false positive and false negatives are not the same, the bank may want to consider a higher threshold to a approve applications

KENT STATE UNIVERSITY

So far, we focused on scenarios where the classification model produces final class predictions such as true or false or default or non-default. In reality, most classification models can additionally generate probability values as their output. For example, the loan default classifier can return the probability of default for each application. Threshold or cut-off will then be applied to convert probabilities or scores to binary classes. For example, if the cut-off value is 0.5, Tom's application will be approved if the probability of default for his loan application (as determined by the model) is smaller than 0.5. While 0.5 may seem to be a logical choice for the cut-off or the threshold value, since the costs of false positive and false negatives are not the same, the bank may want to consider a higher threshold to approve applications.

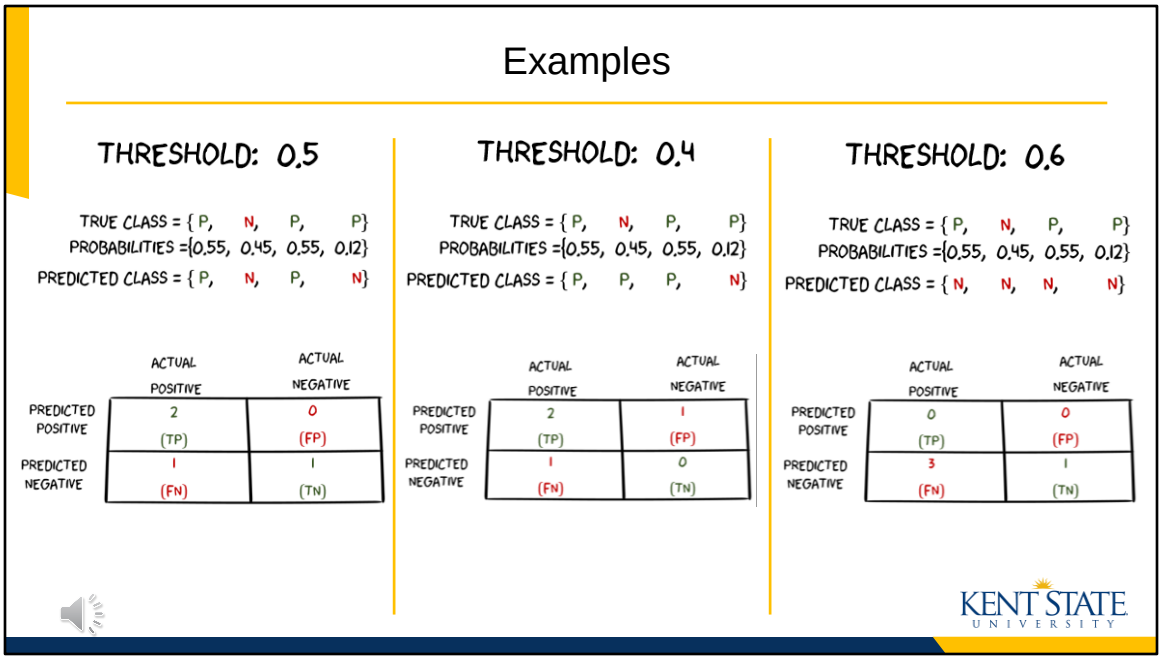
Classification Threshold

- The classification threshold provides an opportunity to trade false positives and false negatives
- Generally, higher the threshold means that we are more picky to consider an observation as positive, which implies lower false positives but higher false negatives and visa versa.



KENT STATE UNIVERSITY

This slide shows how changing the classification decision threshold can be used to create a trade-off between false positive and false negatives. Generally, higher the threshold means that we are pickier to consider an observation as positive, which implies lower false positives but higher false negatives and vice versa. This flexibility is critical in business applications where the cost of false positives and false negatives are substantially different.



Here is a very simplified example to show the effect of changing the threshold. Here we have a vector of true classes consisting of four observations, three positive and one negative. We also have the probability of the positive class from a classification model for each of these observations. More specifically, the model returned the probability of the observations being positive as 0.55, 045, 0.55, and 0.12 for the four observations, respectively. Let's see how the confusion matrix would look like if the threshold is set to 0.5. In this case, the first observation will be considered positive since the probability of being positive, that is 0.55 is greater than the threshold, which is 0.5. Subsequently, the second observation will be considered as negative since 0.45 is smaller than 0.5. The third observation will be positive, and finally, the last observation will be predicted as negative since 0.12 is smaller than 0.5. This suggests that all observations except for the last one will be predicted correctly. The confusion matrix shows two true positives that are the first and the third observations, one true negative that is the second observation, and one false negative that is the last observation.Now, let's reduce the threshold to 0.4. In this case, the first three observations will b predicted as positive since the probabilities are all above 0.45, and the last observation will still be predicted as negative. As such, the second observation is now mistakenly predicted as positive, and therefore we will have a false positive in our confusion matrix. Now, if the threshold is increased to 0.6,

all observations will be predicted as negative as all probabilities are smaller than 0.6. In this case, we will have three false negatives and one true negative.

Classification Threshold

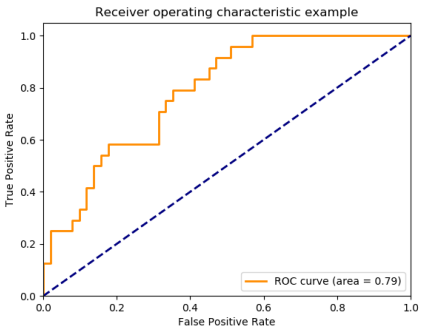
- Accuracy, Precision, Recall, Specificity, and F1 score are performance metrics that are dependent on the choice of threshold.
- This makes it difficult for use them as loss functions to train models, since the measure vary depending on the choice of the threshold.
- So the question is if there exist threshold agnostic performance metrics

KENT STATE UNIVERSITY

Since changing the threshold values results in changes in the number of false positives and false negatives, as you may have guessed, all other metrics that rely on the confusion matrix will also change. This includes classification performance metrics such as Accuracy, Precision, Recall, Specificity, and F1. As such, this makes it difficult to use these performance metrics to train and compare models since the measure varies depending on the choice of the threshold. So you may ask if there is a performance metric that is independent of the choice of the cut-off? Fortunately, the answer is yes.

ROC & AUC

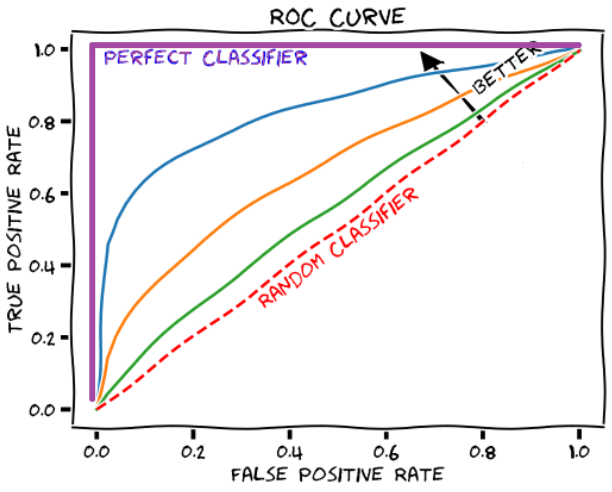
- ROC curves typically feature true positive rate on the Y axis, and false positive rate on the X axis.
- This means that the top left corner of the plot is the “ideal” point - a false positive rate of zero, and a true positive rate of one.
- This is not very realistic, but it does mean that a larger area under the curve (AUC) is usually better.



KENT STATE UNIVERSITY

A ROC curve is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters: True Positive Rate and False Positive Rate. A ROC curve plots True Positive Rate vs. False Positive Rate at different classification thresholds. Lowering the classification threshold classifies more items as positive, thus increasing both False Positives and True Positives.
AUC stands for "Area under the ROC Curve." That is, AUC measures the entire two-dimensional area underneath the entire ROC curve (think integral calculus) from (0,0) to (1,1). AUC provides an aggregate measure of performance across all possible classification thresholds. One way of interpreting AUC is as the probability that the model ranks a random positive example more highly than a random negative example.
AUC ranges in value from 0 to 1. A model whose predictions are 100% wrong has an AUC of 0.0; one whose predictions are 100% correct has an AUC of 1.0.
AUC is desirable for the following two reasons:
AUC is **scale-invariant**. It measures how well predictions are ranked, rather than their absolute values.
AUC is **classification-threshold-invariant**. In other words, it measures the quality of the model's predictions irrespective of what classification threshold is chosen.

AUC



KENT STATE UNIVERSITY

Here is an illustrative example of ROC curves for different classifiers. The random classifier is shown with dashed lines and corresponds to the AUC value of 0.5, which is the area of the triangle under the ROC curve. The further away ROC curves stand from the random curve, the better is the classifier since the area under the ROC curve is larger. A perfect classifier is also shown in this example, which corresponds to the AUC value of 1.

AUC

- AUC avoids the supposed subjectivity in the threshold selection process by summarizing overall model performance over all possible thresholds
- However AUC scores ignore the actual probability values, being insensitive to transformations of the predicted probabilities that preserve their ranks.
- For example, if a fix offset is added to all probabilities, the AUC does not change since the rank order is still the same.
- In case of the Business Bank, AUC works fine if we want to identify top 1000 borrowers that are likely to default. In that case the actual probability values are not important.
- But what if the loans' interest rates are defined based on the probability y of default?

KENT STATE UNIVERSITY

As discussed, AUC avoids the supposed subjectivity in the threshold selection process by summarizing overall model performance over all possible thresholds. However, AUC scores only care about the rank order of the prediction probabilities and ignore the actual probability values. For example, if a fixed offset is added to all probabilities, the AUC does not change since the rank order is still the same.
In some applications, however, we may also need to have an accurate estimate of the actual probabilities. For example, In the case of the Business Bank, AUC works fine if we want to identify the top 1000 borrowers that are likely to default. In that case, the actual probability values are not important. However, if we're going to use default probabilities to determine the interest rates for applicants, then we would need these estimated probabilities to be as accurate as possible.

Log Loss

- Log Loss is another common performance metrics.
- Unlike other classification metrics, the goal of the model is to minimize this value. A perfect model would have a log loss of 0.
- Log Loss takes into account the uncertainty of your prediction based on how much it varies from the actual label. This gives us a more nuanced view into the performance of our model.

EXAMPLE:

THRESHOLD: 0.5

TRUE CLASS = { P, N, P, P}

P(C1) ={0.95, 0.15, 0.45, 0.79}

P(C2) ={0.55, 0.45, 0.55, 0.12}

$$\text{ACCURACY} = \frac{3}{3+1} = 0.75$$

KENT STATE UNIVERSITY

Log Loss is another common performance metrics. Unlike other classification metrics, the goal of the model is to minimize this value. A perfect model would have a log loss of 0. Log Loss takes into account the uncertainty of your prediction based on how much it varies from the actual label. For example, consider a set of four observations a positive, a negative, and two positives. Now consider two different classifiers. Suppose the first classifier gives the following probabilities 0.95, 015, 0.45, and 0.79 for the positive class, and the second classifier gives the following probabilities 0.55, 045, 0.55, and 0.12. Now, if the classification threshold is set to 0.5 what the accuracy of each of the classifiers is? Well, in the case of the first classifier, all predictions but the third one will be correct. In this case, 0.45 is less than 0.5, so the prediction is mistakably classified as negative. For the second classifier, the last prediction will be a false negative. So basically, both classifiers will have an accuracy of 0.75 or 3 over 4. But are they really the same? A closer examination of the probabilities suggests that the first classifier is more reliable since probabilities are much closer to the actual classes. And this is exactly where log-loss comes into the picture as an appropriate loss metric.

Log Loss

- For each prediction, the Log Loss is calculated as

$$-(y\log(p) + (1-y)\log(1-p))$$

- Where *P* is the predicted probability and y is a binary indicator that is 0 if the actual output is false and is 1 if the actual output is true.
- For example, given a class label of 1 and a predicted probability of .25, using the formula above we can calculate the log loss:

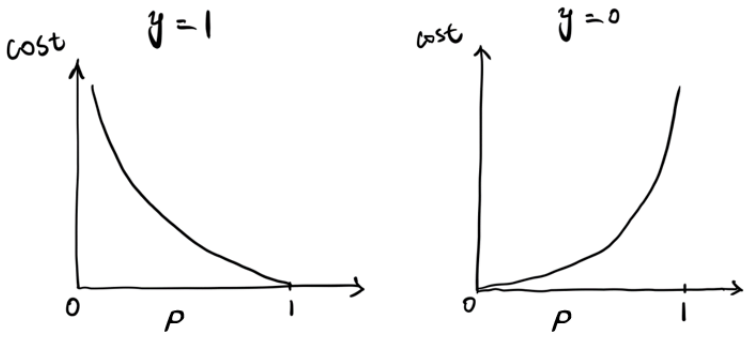
$$-(1\log(.25) + (1-1)\log(1-.25))$$
$$-(\log(.25) + 0\log(.75))$$
$$-\log(.25)$$

- To assess a model, the Log Loss values are calculated for all predictions and then averaged.

For each prediction, the Log Loss is calculated as the negative of the product of y and log of P plus one minus y multiplied by the log of one minus p where P is the predicted probability and y is a binary indicator that is 0 if the actual output is false and is 1 if the actual output is true.
For instance, if the class label is 1 and the predicted probability is 0.25, then the log loss is calculated as -1 log 0.25 plus 1-1 multiplied by log 1-0.25 that is –log 0.25.

Log Loss

- We want to assign more punishment when predicting 1 while the actual is 0 and when predict 0 while the actual is 1



As you can see here, if the label is 1 the log loss decreases with the increase of p and if the label is 0 the log loss increases with the increase of p. That makes sense since we want the probabilities to be as close as possible to 1 when the label is 1 and to be close to 0 when the label is zero.

Log Loss

EXAMPLE: TRUE CLASS = { P, N, P, P}

P(C1) ={0.95, 0.15, 0.45, 0.79}

P(C2) ={0.55, 0.45, 0.55, 0.12}

$$\text{LOG LOSS C1: } \frac{-\log(0.95)-\log(0.85)-\log(0.45)-\log(0.79)}{4} = 0.312$$

$$\text{LOG LOSS C2: } \frac{-\log(0.55)-\log(0.55)-\log(0.55)-\log(0.12)}{4} = 0.978$$

KENT STATE UNIVERSITY

Here is the log loss calculation for our earlier example of the two classifiers. As we can see, the log loss is 0.312 for the first classifier and is 0.978 for the second one. Expectedly the log loss is smaller for the first classifier suggesting that it is a better classifier.

Test Your Knowledge

Suppose we have two classifiers C1 and C2 with the following probability outputs for predicting four observations as shown below.

TRUE CLASS = { P, N, P, P}
P(C1) ={0.95, 0.10, 0.48, 0.80}
P(C2) ={0.59, 0.40, 0.32, 0.54}

Which of the following statements are correct ?

A) We do not need a threshold to calculate classifiers' accuracy

B) The AUC of classifier C1 is higher than C2

C) The Loss Log of C1 is smaller than C2

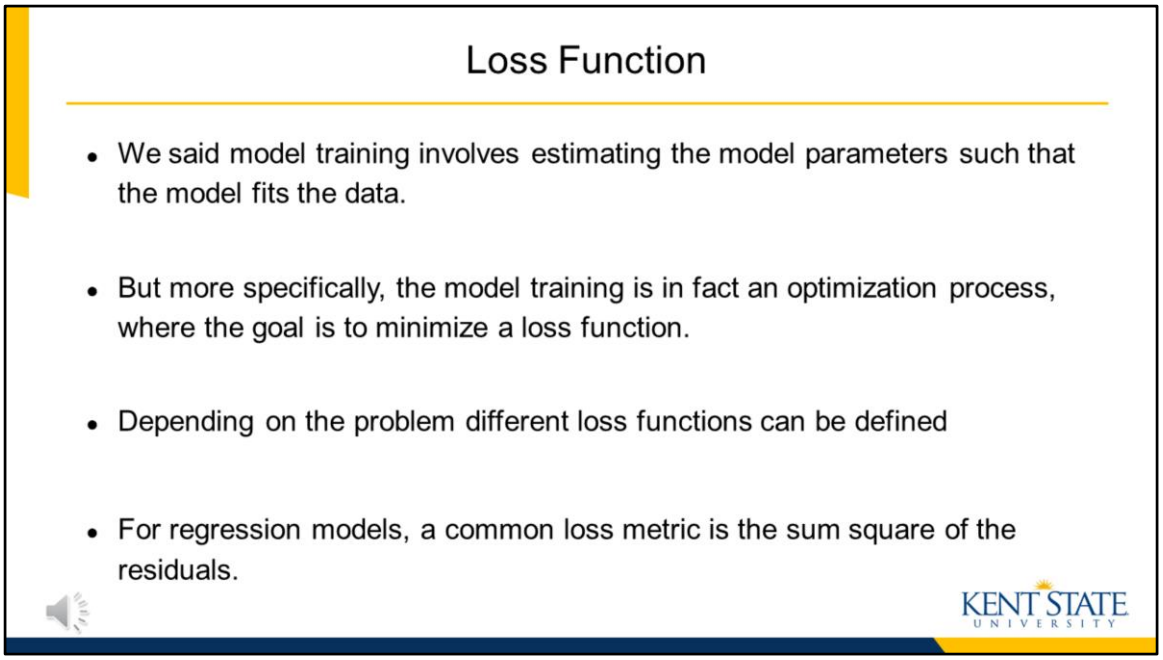
D) The Loss Log of C2 is smaller than C1

- If A selected: Incorrect!! Note that a threshold should be applied to compile the confusion matrix for calculation of accuracy
- If B selected: Incorrect! The rank order of the probabilities are the same for C1 and C2 (i.e. the highest probability is fr the 1st observation, then 4th, then 3rd and finally the second observation) so AUC is the same for C1 and C2.
- If C selected: Correct! C1 is a better classifier and will have a smaller Log Loss
- If D selected: Incorrect! C1 is a better classifier and will have a smaller Log Loss

KENT STATE UNIVERSITY


KENT
STATE
UNIVERSITY
Thank You.

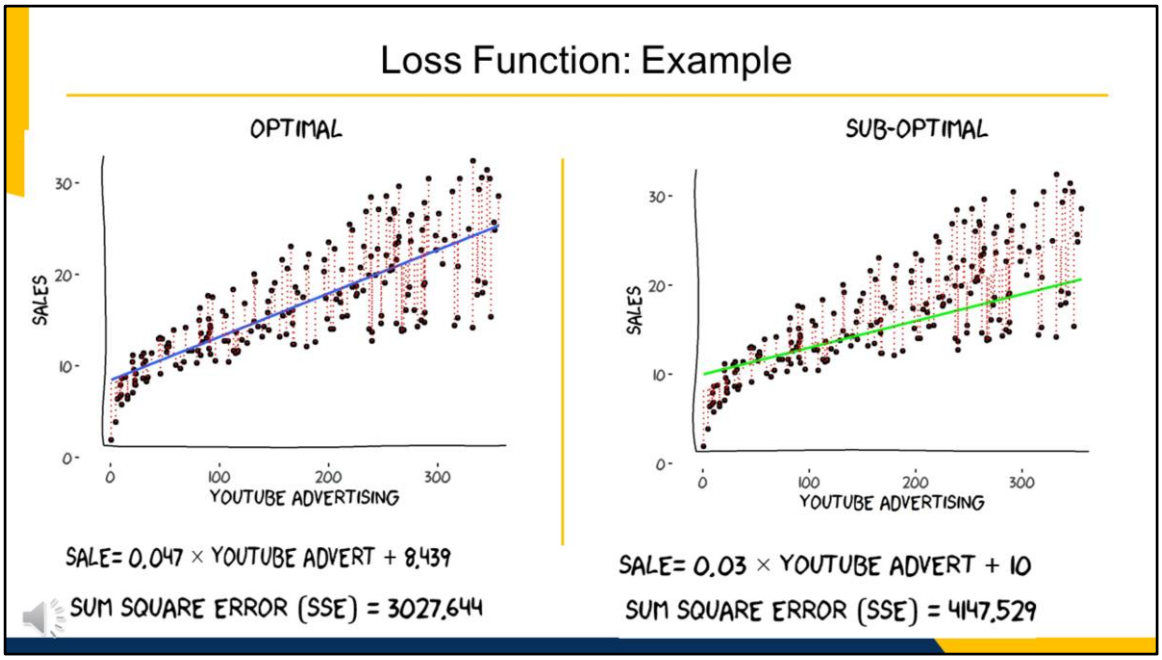
The goal of any machine learning algorithm is to minimize a given loss function. In this presentation, we will introduce the concept of loss function and discuss a number of common loss functions that are used for regression modeling.



Ok, so let's see what a lost function is. Basically, a lost function can be defined as the mathematical formulation of what we are trying to do.

Remember, a machine learning model is nothing but an optimization problem. And when we are optimizing, this should be a function which we call it a cost function, an objective function, or a lost function that we are trying to minimize or maximize. More specifically, you're using the data that is available to us as the training data to find the parameters of the model. These parameters are decided so that the given lost function is minimized.

However, if the goal is to maximize the given function, the function is usually referred to as an objective function. Now, depending on whether we have classification or regression modeling, we have different laws, functions available. In this presentation. We review some of the common lost functions that are used for regression models



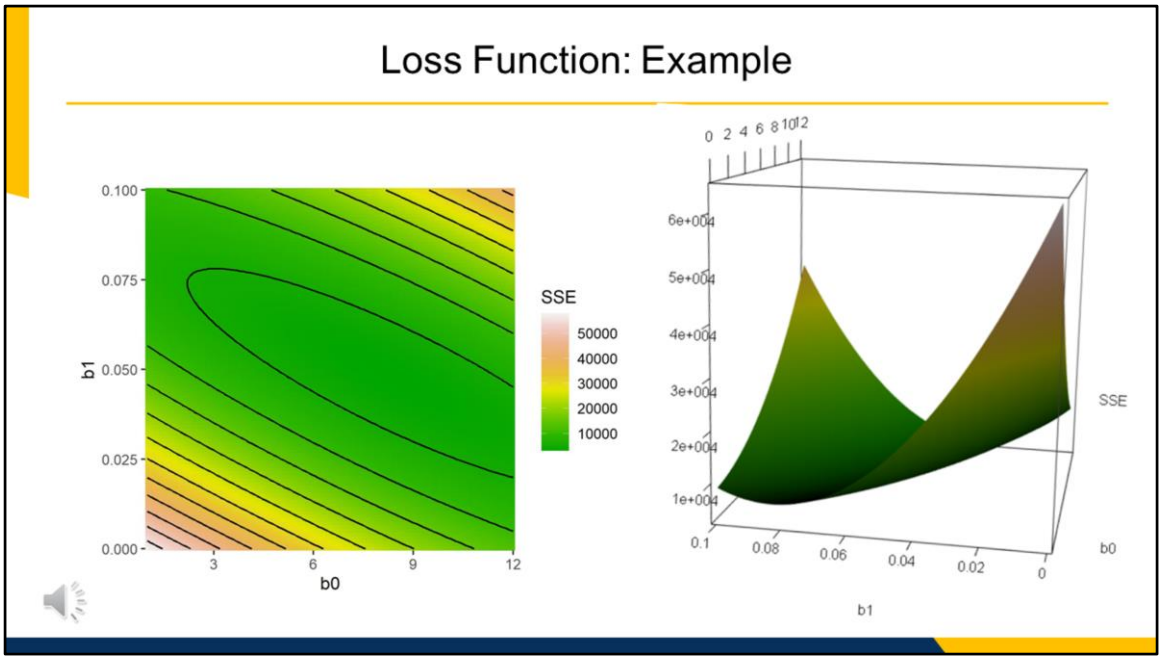
Now, lets consider a regression example here. In this example, we try to understand the relationship between the sales and the money spent on Youtube advertising for a company that produces consumer goods. More specifically, the number of unit of products sold in thousands is our dependent variable shown on the vertical axis and our independent or input variable is the daily budget in dollars that is allocated to Youtube ads. The actual data points are shown as dark circles. The trend and the positive correlation between sales and ads spending make sense as more advertising has shown to translate to higher sales volume. Of course, as you can imagine, this is a simplified example to show the effect of advertising on sales. The real-life scenarios are more complicated since Sale is also a function so many other variables.

So lets say we want to fit a simple linear regression model to model the variations of sales based on Youtube ads spending. If you recall, a linear function is determined by the value of the regression coefficients. In this case we have two coefficients: a fixed constant or b0 and the coefficient for the youtube ads spending or b1. The goal is to fit a line that pass as closely as possible to the actual observed data points. This is exactly our loss function here. More specifically, in simple linear regression models our goal is to minimize the sum of the square of the differences between the line and the actual points. If you recall this difference is also referred to as the regression

residual or error.

But why are we calculating the sum of the square of the error terms as opposed to adding the raw error terms. Well, as you can see, the error terms can be either positive or negative. As such if we add raw residuals in their raw form, they may cancel each other. By taking the residual terms to power two we ensure that all terms are positive, plus the fact that the cost function becomes differentiable, making it easy to find the optimal point analytically. The plot on the left hand side shows the linear model that minimizes the sum square error or SSE. The coefficient values corresponding to this model are 0.047 for b1 and 8.439 for b0. Now, lets see what would happen if the model's coefficient were 0.03 and 10 respectively. As shown on the Figure on the right, the residuals is increased for most data points above the line and is decreased for most data points below the fitted line. However, the overall effect is that the SSE is increased.

the dependent variable is



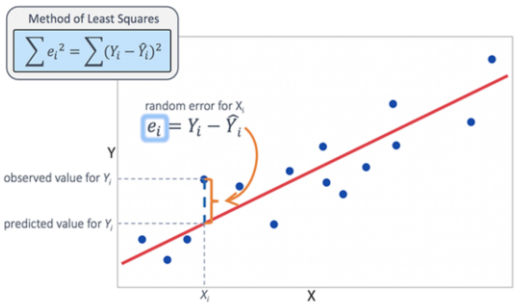
Here you can see how SSE varies with respect to changes in the value of the models' coefficients. More specifically, the left graph shows the value of b0 and b1 as the x and y-axis, while the SSE is color codded. As we saw previously, SSE is minimized when b1 is 0.047 and b0 is 8.439. Deviation from this optimal value results the SSE to be increased. Similarly, the right plot shows the 3D plot of SSE versus b0 and b1 where similar patterns can be seen.

Loss Function: RMSE

- Root Mean Square Error, RMSE is also times used as a metric for evaluating the fitness of a regression model

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

- Note that minimizing RMSE is the same as minimizing SSE



KENT STATE UNIVERSITY

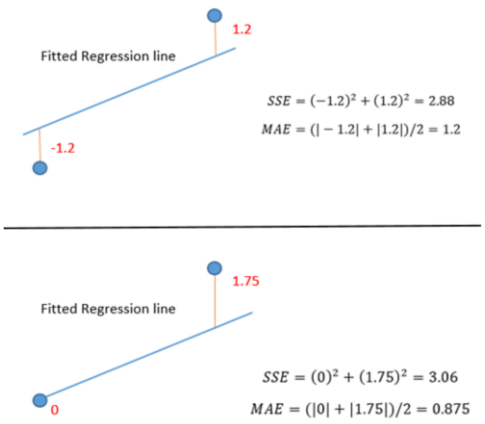
One drawback of using SSE as the loss function is that SSE always increases as we have more data points. As such the actual value of SSE does not say much about the model goodness. Moreover, SSE is expressed in squared units of the dependent variable. For example, if the dependent variable is the height of newborn babies, expressed in inch. The SSE of the model is expressed in inch squared which does not have any meaning. To address these two issues, the Root Mean Square Error or RMSE is normally used as the fitness function. RMSE is calculated by dividing SSE by the number of data points, n, and then to calculate the square root of the results. Since the square root is a monotonic function, the coefficients that minimize SSE are the same as those that minimize the RMSE. In other words, in our previous example, the regression coefficients would be the same if SSE or RMSE are used as the loss function.

Loss Function: MAE

- Another loss function used for regression models is the Mean Absolute Error (MAE).

$$\text{MAE} = \frac{1}{n}\sum_{j=1} |y_j - \hat{y}_j|$$

- The main drawback of MAE as a loss function is that this loss function is not differentiable at 0.

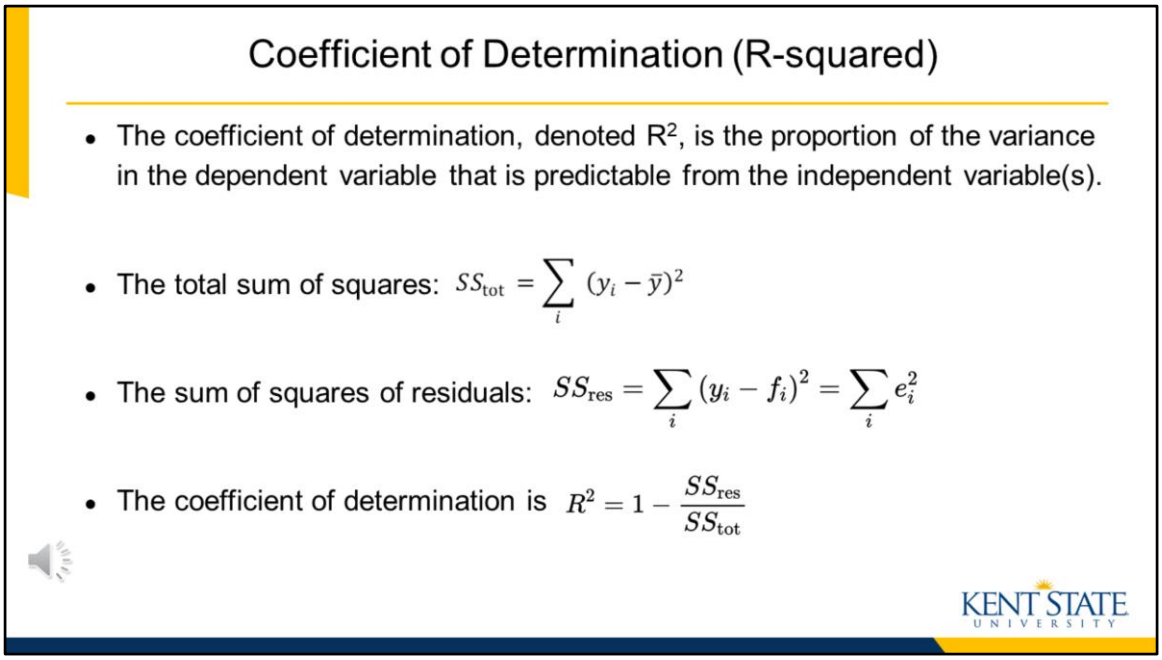


KENT STATE UNIVERSITY

So far we have used the square of the distance of data points from the regression line as measure of goodness of the fit. Since the error terms or residuals are squared, the error effect is going to be amplified. Therefore, when we use the square of the residuals as a part of the loss function, our model try to fit a line that is not too far from data points since the effect of residuals will be further amplified when they are taken to the power two. For example, when using the square of the error terms, a regression line that is 1.2 units distanced from each of the two data points is considered to be a better fit when is compared to a regression line that goes exactly through one data point but is 1.75 units distanced from the other data point. This is because in the firs scenario the residuals are 1.2 for both data points, the square error terms are 1.44 each and the SSE that is their sum is 2.88 . In the second case, the residual is zero for the first data points since the regression line pass through it, and is 1.75 for the second one. The square of 1.75 is 3.06 and therefore the SSE is 3.06 plus zero that is 3.06.

However, the business problem that we are trying to solve may not necessarily have such requirements. In other words, in our business problem we may prefer to have one data point that is 1.75 units distanced from our prediction than two data points each distanced 1.2 units from our prediction. This is where the Mean Absolute Error

or MAE can be used as a loss metric. As its name implies, MAE is the average of the absolute values of the error terms or residuals. While, it can be useful in many scenarios, the main drawback of using MAE is that the absolute value operator makes the loss function not be differentiable at 0. As such, analytical optimization methods can not be easily applied to find the minimum of the loss function. Instead, heuristic optimization methods should be used if MAE is used as the loss function.



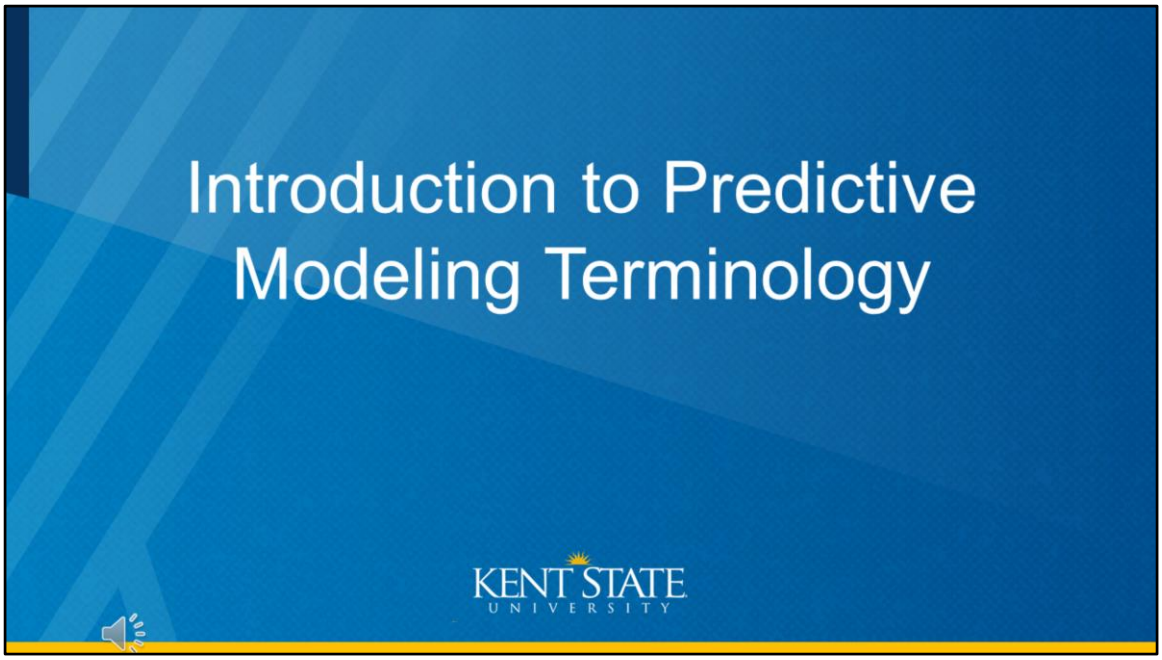
Coefficient of Determination or R-squared is another common metric used to judge the goodness of fit for regression models. The coefficient of determination is interpreted as the proportion of the variance in the dependent variable that is predictable from the independent variable. The coefficient of determination is the square of the correlation (r) between predicted scores and actual scores; thus, it ranges from 0 to 1.

In general, the higher the R-squared, the better the model fits your data. R-squared can be calculated as one minus Sum Square of Regression or SSres that is the same as SSE divided by the total sum squares or SStot. One advantage of the R-squared is that the metric does not have any unit, and therefore it can be used to compare different models. Now you may ask if low R-squared values always a problem. Well, the answer is no. Some fields of study have an inherently greater amount of unexplainable variation. In these areas, your R2 values are bound to be lower. For example, studies that try to explain human behavior generally have R2 values of less than 50%. People are just harder to predict than things like physical processes.

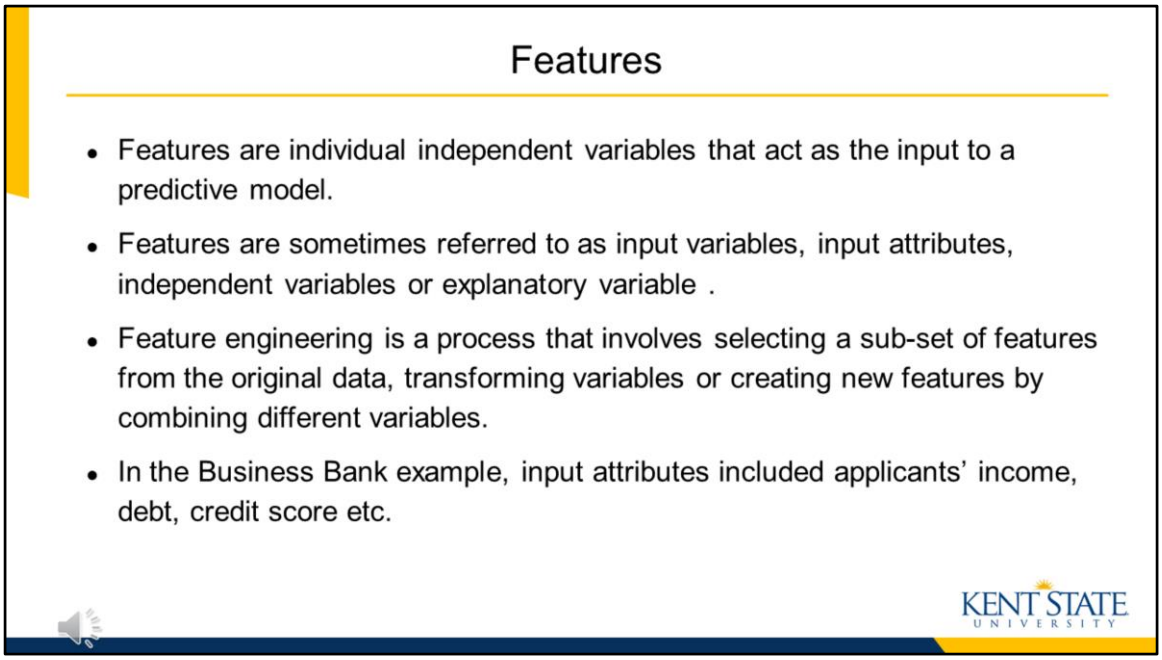
Finally, note that it is a good habit to calculate various performance metrics and loss or objective functions when comparing different regression models as each metrics

provide a unique insight about the goodness of fit.

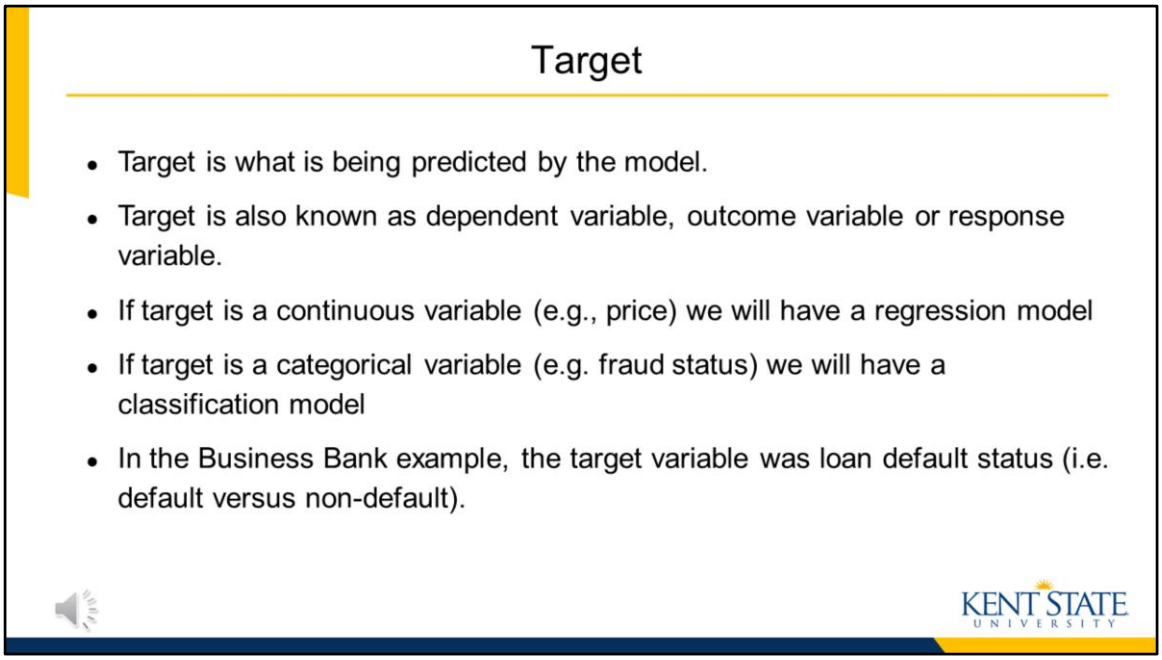

KENT
STATE
UNIVERSITY
Thank You.



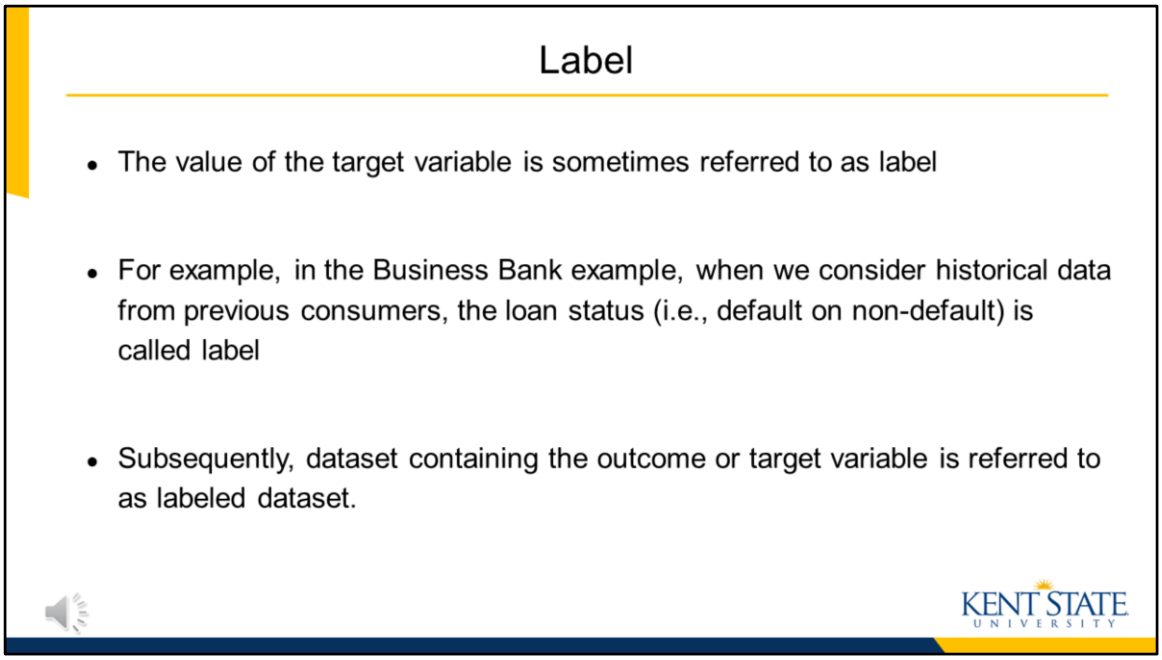
In this presentation, we will introduce some key definitions in predictive modeling.



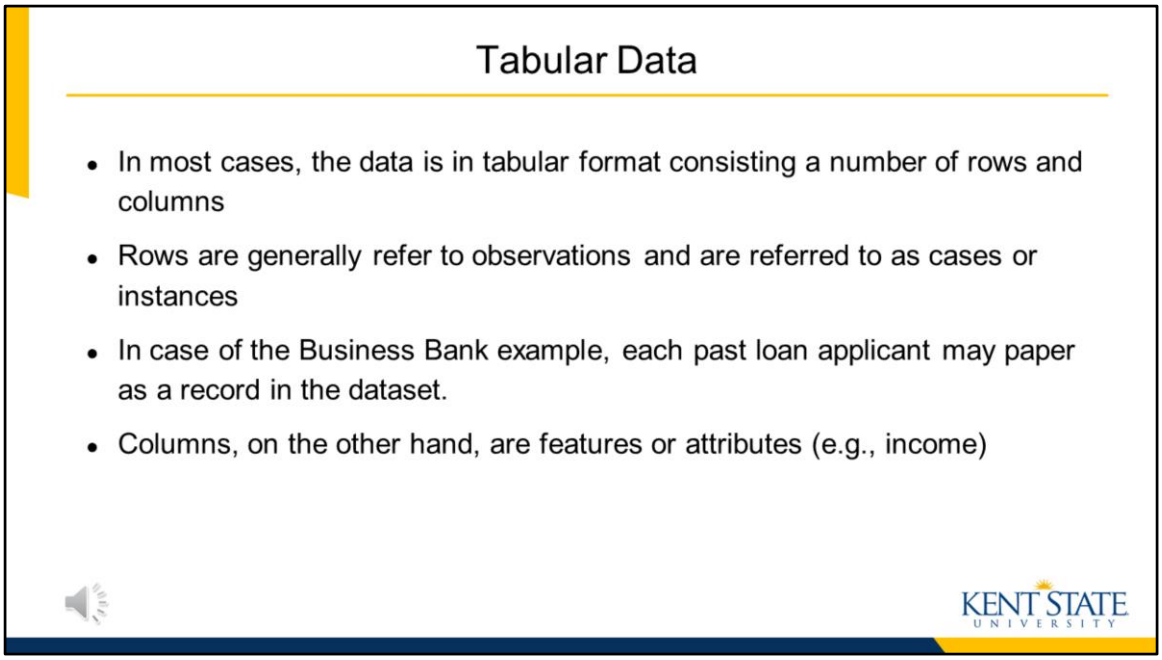
Let's start with features. Features are individual independent variables that act as the input in your system. Prediction models use features to make predictions. More simply, you can consider one column of your data set to be one feature. Sometimes these are also called attributes, and the number of features are called dimensions. New features can also be obtained from old features using a method known as 'feature engineering'. Feature engineering is a process that involves selecting a subset of features from the original data, transforming variables, or creating new features by combining different variables.



The “target variable” is the variable whose values are to be modeled and predicted by other variables. It could be the individual classes that the input variables may be mapped to in case of a classification problem or the output value range in a regression problem. Target is also known as a dependent variable, an outcome variable, or a response variable.



The value of the target variable is sometimes referred to as a label. For example, in the Business Bank example, when we consider historical data from previous consumers, the loan status is called label

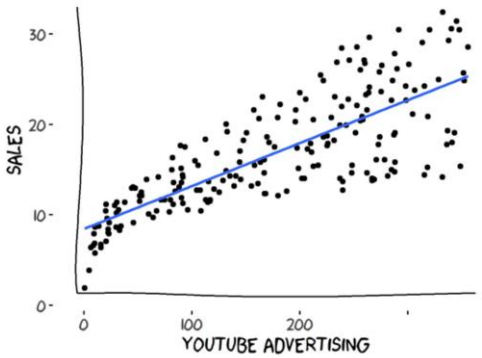


In most cases, the data is in a tabular format consisting of a number of rows and columns. Rows generally are observations and are referred to as cases or instances In the case of the Business Bank example, each past loan applicant can appear as a record in the dataset. Columns, on the other hand, are features or attributes. This includes, for example, customers' income, their debt, or their credit score.

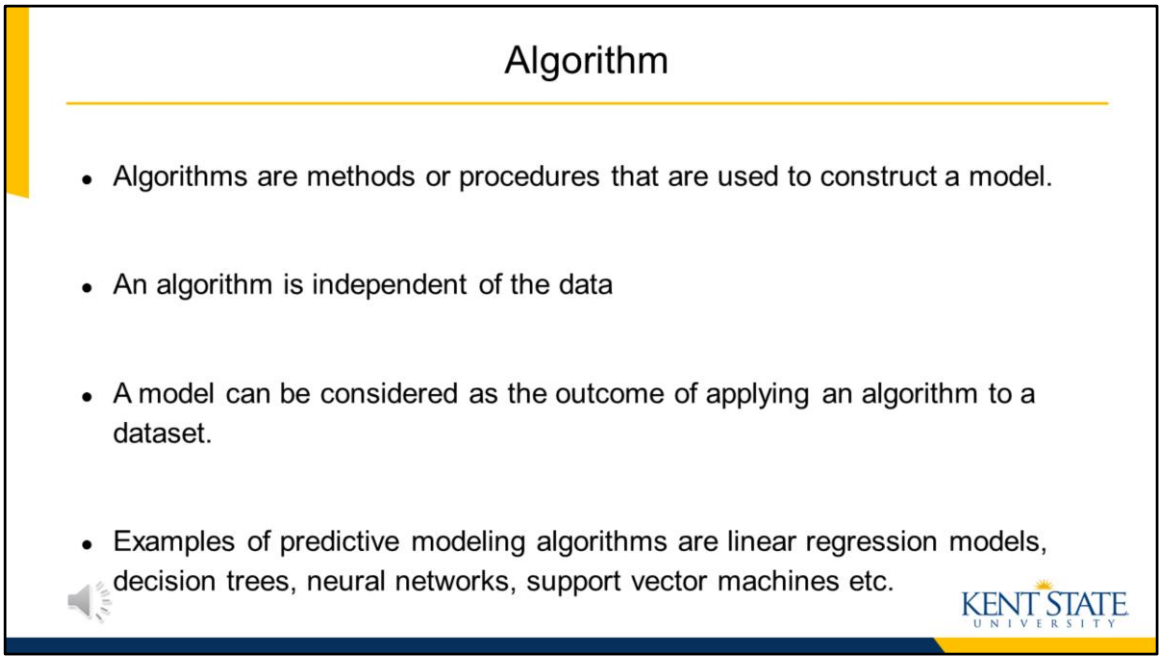
Model

- A model can be thought as a function that map the input variables to the target.

SALE= 0.047 × YOUTUBE ADVERT + 8.439



A predictive model can be described as learning a function (f) that best maps input variables (X) to an output or target variable (Y).This is a general learning task where we would like to make predictions in the future for Y, given new examples of input variables (X).We don't know what the function (f) looks like. If we did, we would use it directly, and we would not need to learn it from data. However, we can define the form of the function and use data to find the parameters of the function. For example, as you will see, a linear regression function is defined by choosing a set of coefficients associated with input variables and a fixed bias term.



Algorithms are methods or procedures that are used to construct a model. An algorithm is independent of the data. More specifically, a model can be considered as an outcome of applying an algorithm to a dataset.

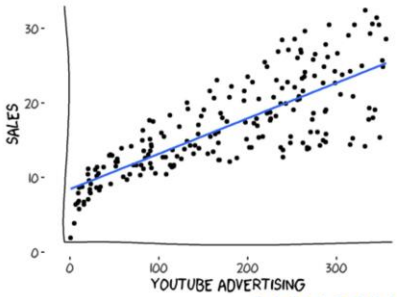
Model Parameters

- If you recall, a model is a function that map input variables to the target variable
- A model adapt itself to the data (i.e. learns from data) by changings its parameters.

SALE= 0.047 × YOUTUBE ADVERT + 8.439

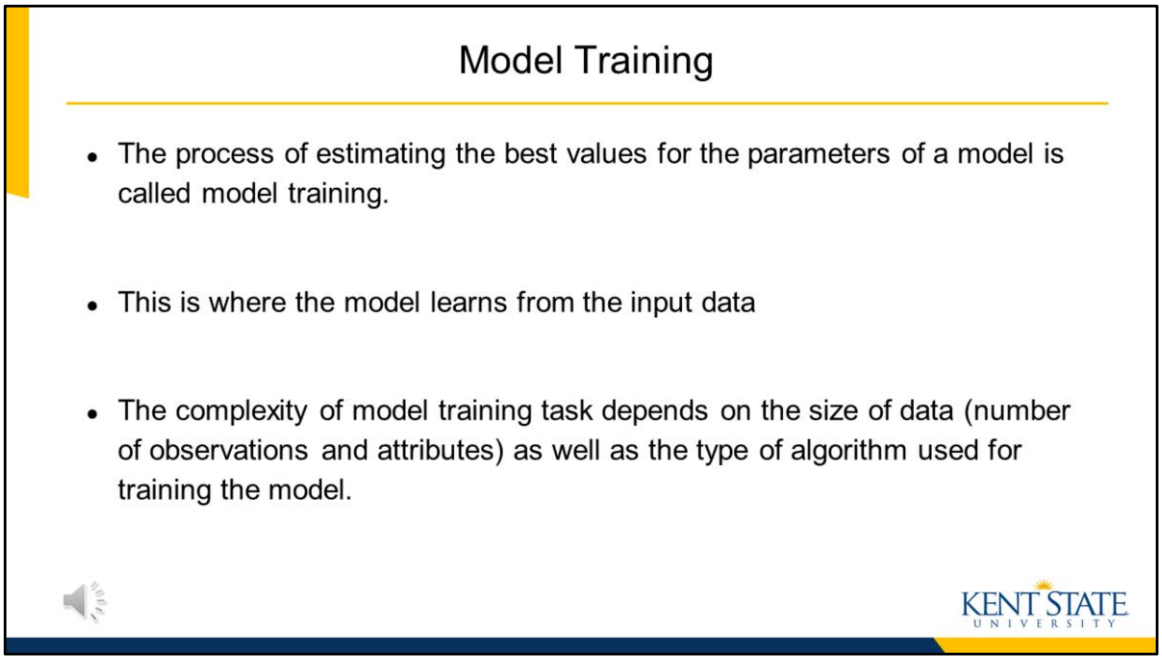
PARAMETER 2

PARAMETER 1

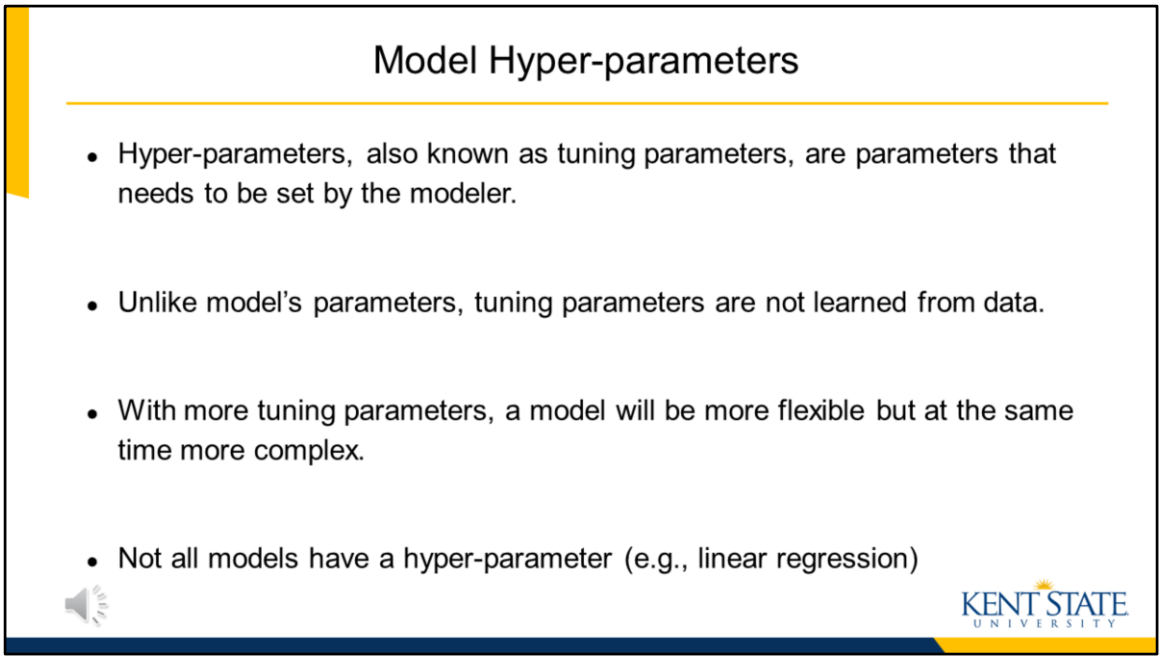


KENT STATE UNIVERSITY

A model parameter is a configuration variable that is internal to the model and whose value can be estimated from data. Model parameters are required by the model when making predictions. Also, they are estimated or learned from data and are often saved as part of the learned model. Parameters are key to machine learning algorithms.

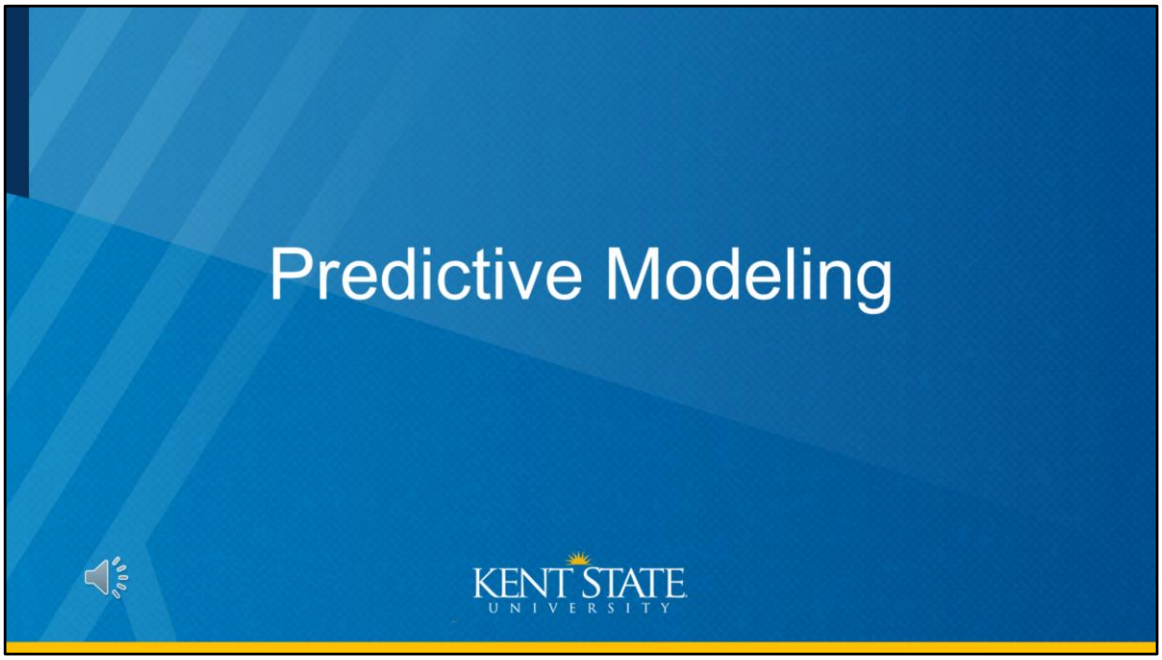


The process of estimating the best values for the parameters of a model is called model training. This is where the model learns from the input data. Predictive algorithms learn from examples and, if you have relevant and high-quality data, the more examples you provide, the better it will be at finding patterns in the data.

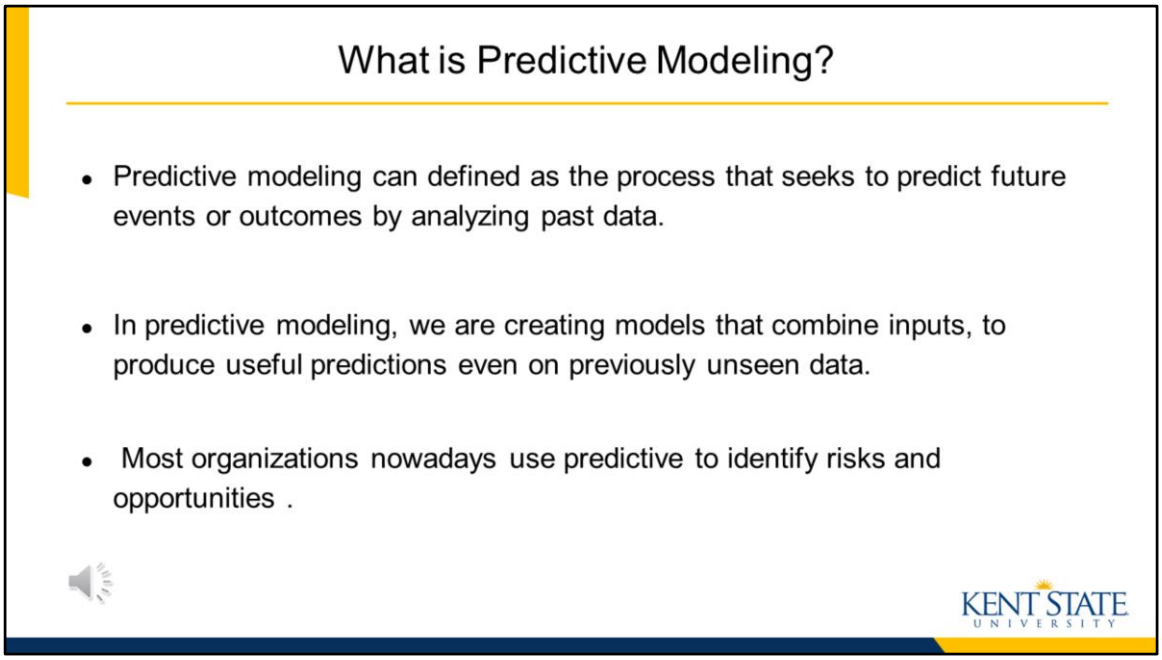


Although the terms parameters and hyper-parameters are occasionally used interchangeably, we are going to distinguish between the two. Parameters are properties the model learn during training. For linear regression, these are the weights and biases;. Hyper-parameters, on the other hand, are properties that must be set before training. Hyper-parameter optimization is the process of finding the best possible values for these hyper-parameters to optimize your performance metric. To do this, we train a model for different combinations of values and evaluate which combination is the best.


KENT
STATE
UNIVERSITY
Thank You.



In this presentation, we will describe predictive modeling and discuss few examples of predictive modeling and their business applications.



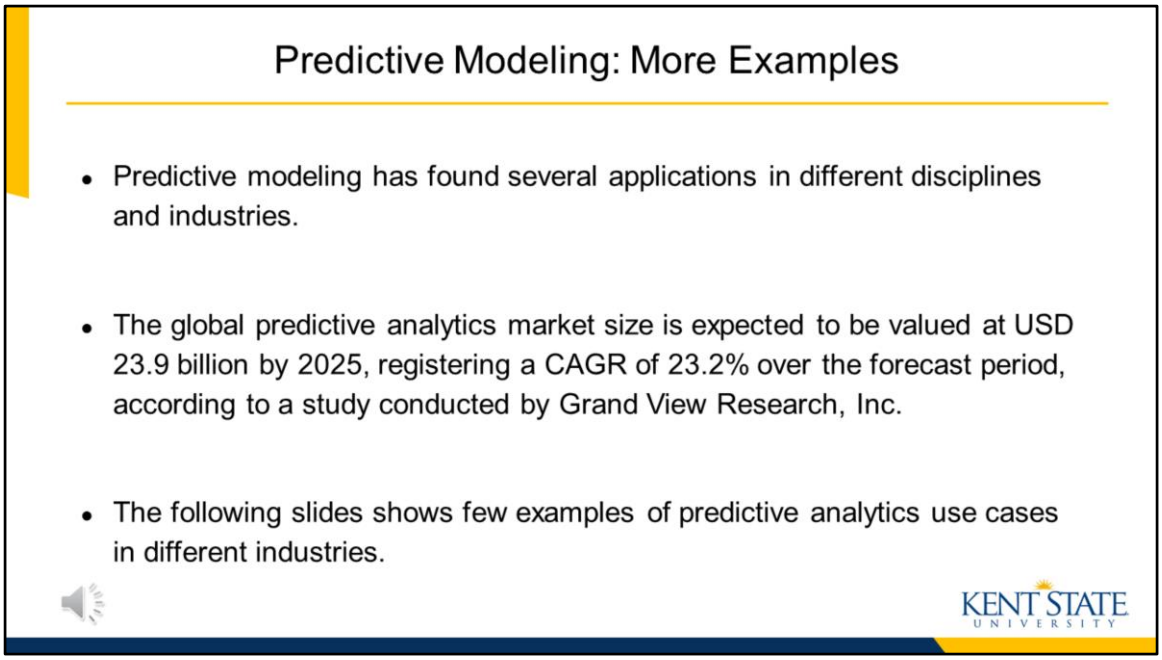
Predictive modeling is a commonly used statistical technique to predict future behavior. Predictive modeling solutions are a form of data-mining technology that works by analyzing historical and current data and generating a model to help predict future outcomes. In predictive modeling, data is collected, a statistical model is formulated, predictions are made, and the model is validated (or revised) as additional data becomes available. For example, risk models can be created to combine member information in complex ways with demographic and lifestyle information from external sources to improve underwriting accuracy. Predictive models analyze past performance to assess how likely a customer is to exhibit a specific behavior in the future.

Let's consider an example. In this example, we have the Business Bank, which is a community bank in North Ohio that specializes in serving small businesses. Like most other community banks, Business Bank derives its sources of funds from and operates exclusively in northeast Ohio. The bank is proud to have served the local community for more than 50 years.
The bank offers the entire standard banking services, including checking, savings, loans and mortgages, and safe deposit boxes for both consumers and business customers. The bank believes that, compared to their larger counterparts, they can better serve the community since they make all lending decisions locally by people who understand the unique challenges and financial needs of the local market. Loans to small businesses are, therefore, one of the primary sources of revenue for the bank.

Now consider Tom. Tom is a local restaurant owner who wants to expand his business. However, Tom does not have the required capital to do so. Tom has heard about the Business Bank from a friend, so he decides to apply for a business loan. If Toms' loan application is approved, there will be two outcomes: either Tom will repay the loan in full, which in that case the bank will make a profit, or Tom will default on his loan, which implies financial losses for the bank.

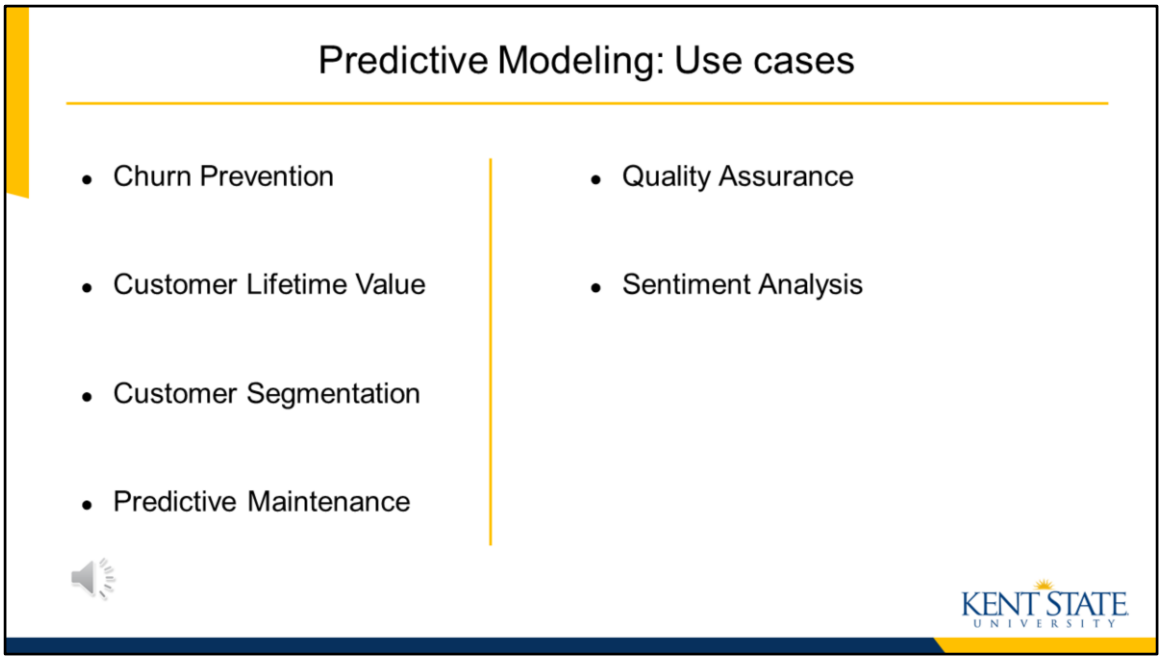
Tom's loan application is now with Jeff, who is a loan officer with the Business Bank. Jeff's job is to determine Tom's creditworthiness. In other words, he needs to decide to approve or decline Tom's loan application. But how can Jeff do that? Of course, Jeff can make underwriting decisions based on his intuition. But making decisions merely based on guts feelings will most likely be sub-optimal. In addition, how can the bank justify their decision to financial regulators? As you guessed, Jeff needs to use a data-driven approach for making the underwriting decision. Let me explain. Obviously, Tom is not the first borrower who has applied to the Business Bank. Remember, the bank has been serving the community for the past 50 years. So it seems logical to use data from the previous borrowers to asses Tom's credit risk. Well, this is exactly what a predictive model does.

As its name suggests, a predictive model takes a number of input attributes and generate a prediction as an output. In the case of a loan underwriting model, input attributes could include things such as the applicants' income, current debt, and credit history, to name a few. These attributes will determine if an applicant has the means and resources to pay off their loan. The output of the model will be a risk indicator, such as the probability of loan default. With this model, Jeff is now able to make a more informed decision about Tom's application. Note that the model will not be correct at all times. As well said by the British statistician George Box, all models are wrong, but some of them are useful. In the case of a loan underwriting model, it is possible that the model denies the loan application of a creditworthy applicant or to approve a loan that will ultimately default. Since the bank revenue is closely tied to the performance of the underwriting model, the analytics department in the Business Bank worked over the years to improve the accuracy of the model.
Now that you get an idea about predictive models, try to think about other examples where predictive models are used to derive decisions.



As you can imagine, the possibility of forecasting future outcomes can provide organizations with a significant competitive advantage and that's why you see a significant growth of predictive analytics tools and frameworks over the past few years. This is especially relevant considering the technological advances that we have seen over the recent years. More specifically the cost of collecting, storing and processing large volumes of data has reduced significantly over the past few years. This allows organizations to collect, store, and process massive volumes of data in a cost-efficient way. Note that the accuracy of the predictive models very much depends on the volume and the quality of the input data, because at the end of the day, the algorithms are simply trying to find meaningful patterns from the input data to predict future outcomes.

By 2022 the predictive analytics market is expected to reach nearly 11 billion dollars in annual revenue as an increasingly large number of businesses make use of predictive analytics techniques for everything from fraud detection to medical diagnosis. In the next few slides, we will discuss a few examples of predictive analytics use cases in different industries.



Here are few uses cases of predictive modeling

The first example is customers'' Churn Prevention.
When a business loses customers, it needs to bring new customers in to replace the loss in revenue. That obviously can get very expensive, because acquisition of a new customer is usually much more expensive than existing customer retention.
Predictive analytics help to prevent churn in companies' customer base, by identifying signs of dissatisfaction among customers, and identifying those customers or customer segments that are at the most risk for leaving. Using that information, companies can then make the necessary changes to keep those customers happy and protect their revenue.

Next we have Customer Lifetime Value prediction.
One of the more difficult things to do in marketing is to identify those customers that are going to spend the most money, in the most consistent way and over the longest period of time. This kind of insight allows companies to optimize their marketing to increase their share of that segment of the business, and gain those customers that

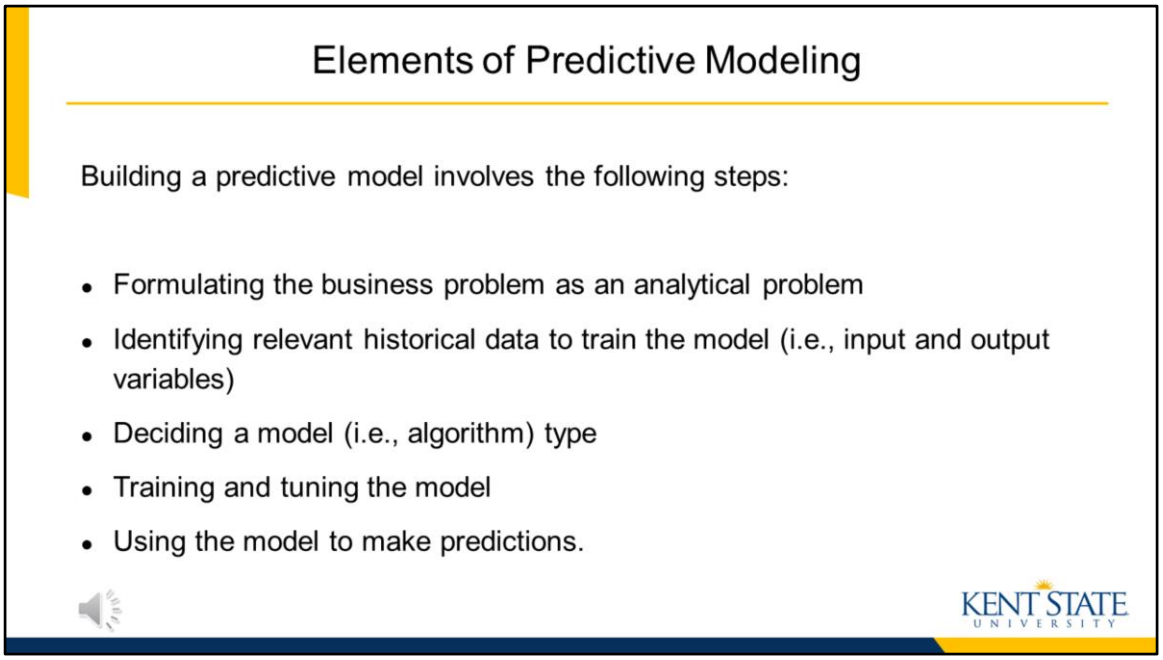
will have the greatest lifetime value to the company.

Then we have Customer Segmentation
Note that different companies define their markets differently, and segment their markets according to those aspects that offer the most value to their particular industry or products. A good use of predictive analytics is to identify target markets based on real data and indicators, and further identify the segments of those markets that are most receptive to what a company offers. This same data can also help to identify segments and potentially even entire markets that a company even did not realize that existed

Predictive Maintenance is another great example of predictive analytics use cases. For companies with a major investment in infrastructure and equipment, the ability to manage that capital cost is critical. By analyzing metrics and data related to the lifecycle maintenance of technical equipment, companies can predict both timelines for probable maintenance events and upcoming capital expenditure requirements, allowing them to streamline their maintenance costs and avoid critical downtime.

Then we have Quality Assurance
Quality control is key to not just the customer experience, but also operational expenses as well. Over time, inefficient quality control will affect customer satisfaction, buying behaviors, and ultimately impact revenues and market share. Poorer quality control also leads to more customer support costs, warranty issues and repairs, and less efficient manufacturing. Good predictive analytics, however, can provide insight into potential quality issues and trends before they become truly critical issues.

Next we have Sentiment Analysis.
Well It's very difficult to be everywhere at all times, especially in the online world. Likewise, capturing and reviewing everything that's said about a company or organization is virtually impossible. However, by combining web search and crawling tools with customer feedback and posts, companies can create analytics that give a picture of organization's reputation within a given market and demographics, and provide proactive recommendations as to the best ways to enhance that reputation.



Regardless of the types of predictive models in place, the process of predictive model deployment follows the same steps.

The first step is to define the business problem that we are working on in a format that can be formulated as an analytics project. At the end of the day, an analytics project can be considered as an optimization problem. Our objective function in the optimization problem should be in line with the business problem that we are trying to solve. For example, for predicting loan defaults, our mathematical objective function could be the number of applicants that are misclassified. This could be creditworthy applicants that are denied or applicants who will eventually default and are approved. The objective function in our classification model could be then to minimize the overall misclassification rate. However, as we discussed, one may argue that the cost of false negatives and false positives could be different and therefore, the objective function should assign different weights to each type of error. In general, in it is very important to formulate our optimization problem such that it is in line with the needs and requirements of the business. Effective communication between the analysts and project sponsor is, therefore, critical during the problem formulation phase.

Once the problem and project's goals are defined, relevant data for the modelling task should be identified. Mixing and merging data from as many data sources as possible is what makes a data project great, so look as far as possible. Ask the data and IT teams for the data that's available or open up your private database and start digging through it to understand what information the organization has been collecting. You should also look for open data. The Internet is full of datasets to enrich what you have with additional information. For example, census data will help you add the average revenue for the district where your user lives or OpenStreetMap can show you how many coffee shops are on a given street. A lot of countries have open data platforms (like data.gov in the U.S.).

In addition to sourcing data, you will likely need to clean and transform your data to make it suitable for your modeling task. Data cleaning is the process of preparing data for analysis by removing or modifying data that is incorrect, incomplete, irrelevant, duplicated, or improperly formatted. This data is usually not necessary or helpful when it comes to analyzing data because it may hinder the process or provide inaccurate results.

At this point, you should have a good understanding of the problem and the data that you have available for the modeling task. This should help you plan and chose the right modeling strategy for your task. The structure of the data will dictate what tools and analytic techniques can be used so as the nature of the problem that you are trying to solve. For example, some predictive algorithms may not support your data type (for example, if you have a mix of numeric and categorical variables) or maybe too complex for the task in hand. In general, your goal should be to keep the overall complexity of the modeling project as low as you can. This will help you better optimize, maintain, and calibrate the model. In this course, we will learn about a few popular predictive modeling algorithms, and you will understand the pros and cons of each method.

Once the modeling framework is decided, you will then need to use the data to train and optimize the model. As we will see, model training involves splitting available data into training and testing sets and optimizing and tuning the models. Finally, once you are happy with your model, the model can be used to make predictions.


KENT
STATE
UNIVERSITY
Thank You.