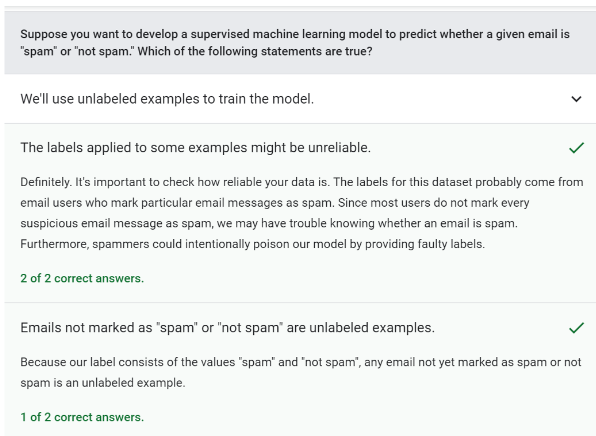
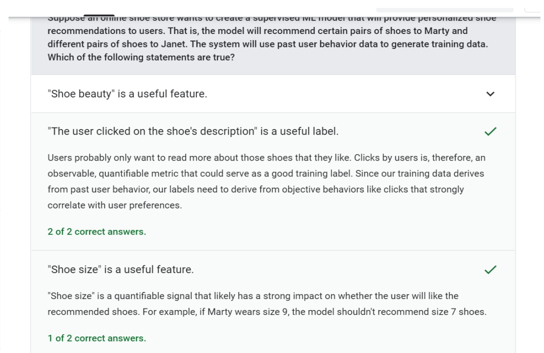
# <u>REPORT- ML CRASH COURSE</u>

## 1. FRAMING:

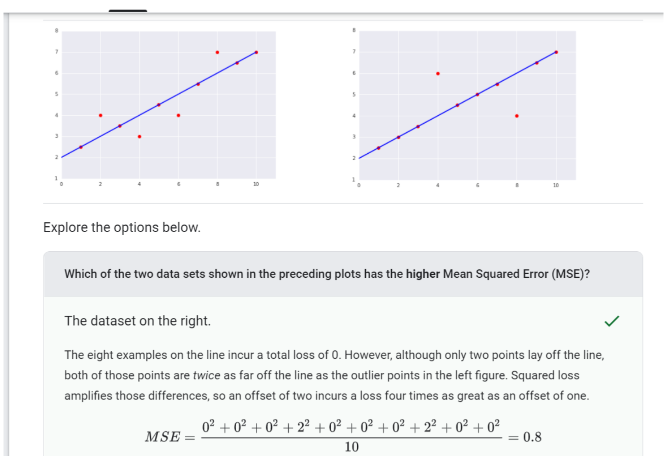
In this section, we learnt the common Machine Learning terminology that will be frequently used henceforth.



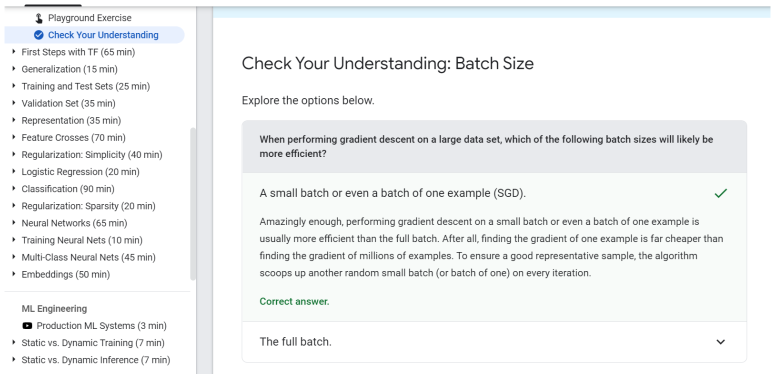

## 2.DESCENDING INTO ML:

Here, we brush upon the basic concept of Linear regression. We also learn about a few loss functions like L2(squared) loss and mean squared error(MSE). Minimising loss functions makes the model more accurate while training.




## 3.REDUCING LOSS:

We randomly guess the bias and weights of the linear equation and compute loss function. Then we accordingly change them(using gradient descent) until we find the optimised parameters having minimum loss value. Then the model is said to be converged. We use learning rate as a hyperparameter to go to the next point. Finding a suitable learning rate is very important. There are two types of gradient descent- Stochastic and batch. stochastic gradient deals with only one example. But a batch considers a set of examples.




**4. FIRST STEPS WITH TF:**

This section deals with the introduction to TensorFlow- open source code library to develop ML models. The term epoch means the no. of times the model runs through the dataset. There are 3 hyperparameters we need to tune to create a model that converges efficiently. They are learning rate, epochs, batch size. There are no hard rules to tune these parameters. It purely depends on the dataset. We make use of correlation matrix to understand which feature is best for prediction of the label.

**5. GENERALISATION:**

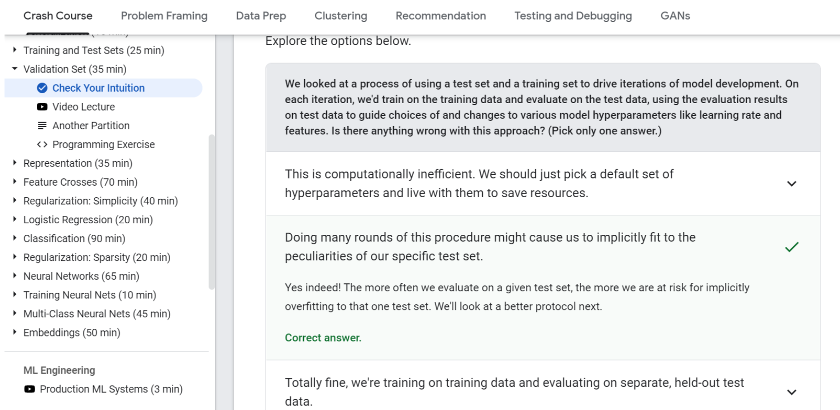
We learnt about overfitting models. An overfitting model works very well with the training data with very little loss, but fails to adapt properly to the new set of data given for prediction. The key is to make the model quite simple and as generalised as possible without hardcoding around the peculiarities in the training set. A way is to divide the dataset into 2 subsets- training set and test set. But we need to make sure the subsets have randomly chosen examples and that the test set is large.

**6. TRAINING AND TEST DATASETS:**

We need to slice the dataset into training and test dataset. We need to make sure that the test data is representative of the whole set. Also we must confirm that we do not train our model on the test dataset itself.

**7. VALIDATION SET:**

Repetitive training,testing and tweaking the model on the dataset will lead to overfitting of exceptions in the dataset. To avoid that, we can introduce another subset- Validation set. Testing the model after making any changes to the hyperparameters can be done on validation set, leaving the test data untouched. Once the validation loss almost equals testing loss, we can say that we have avoided overfitting.

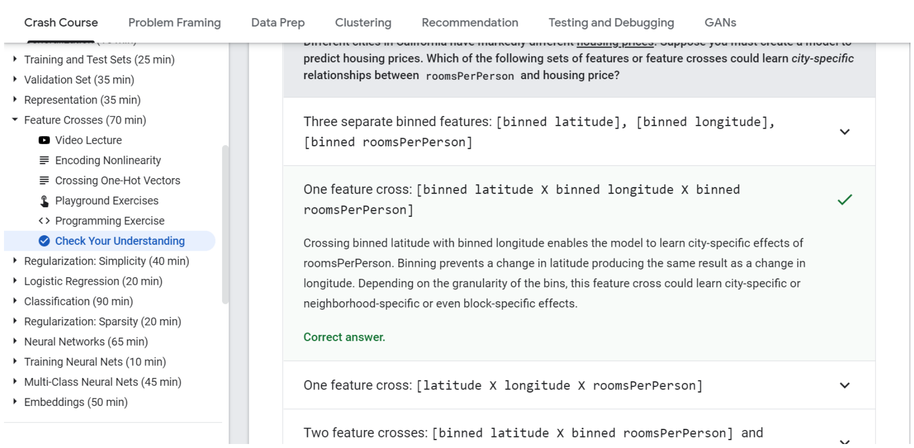



**8. REPRESENTATION:**

This section deals with Feature engineering. Extracting useful features from the dataset and modifying them into a form which we can use for training the model is very important. It takes up most of the time. Mapping numerical features is simple. We can simply scale it down or up according to our convenience. But categorical variables need some encoding to be done. Common practices are One Hot Encoding and Multi Hot Encoding. It creates binary feature vectors. If there are many categories then sparse representation is used. We can use binning for features that don't linearly depend on the label.

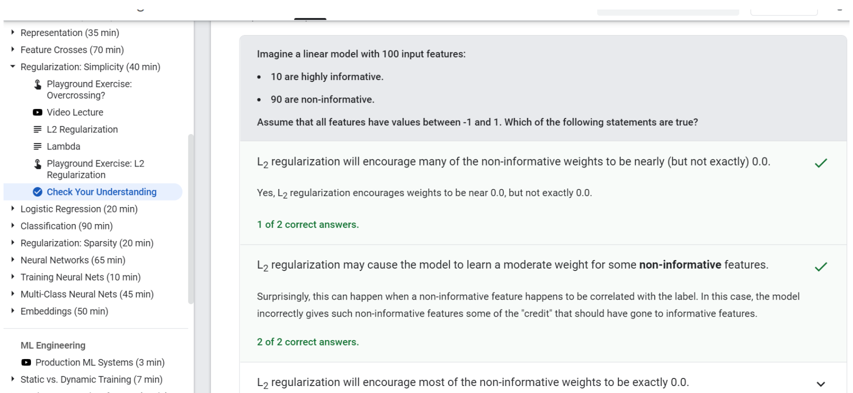
**9. FEATURES CROSSES:**

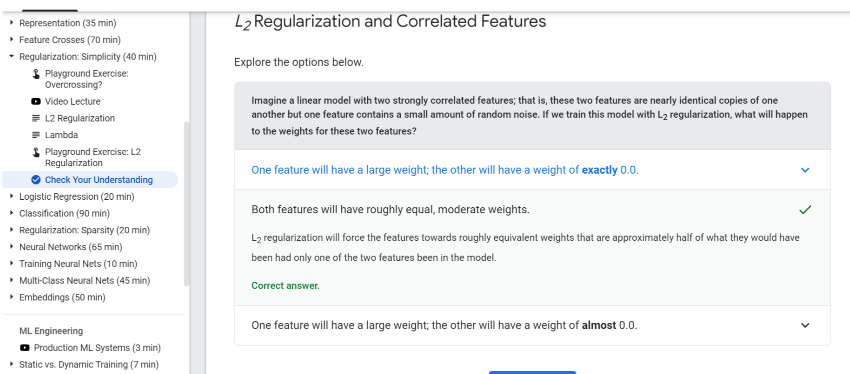
When we make new synthetic features by crossing 2 features, it is known as feature crosses. They help in bringing non linearity in linear models. Often we cross one hot encoded features.




**10. REGULARISATION FOR SIMPLICITY:**

We do regularization to deal with overfitting. Instead of minimising only the loss, we minimize loss and complexity of the model. This is known as structural risk minimization. Regularization term measures the model complexity. In L2 regularization approach, we measure complexity as sum of squares of the weights. It is added with cost function. We use a hyperparameter lambda to tune the complexity of the model. Lambda must not be higher else it will lead to underfitting and improper predictions. Note that when we add regularization terms, training loss will increase but test loss will drop because we added new term. And also the weights decrease and come closer to zero. It tries to decrease the large weights.




**11. LOGISTIC REGRESSION:**

Logistic regression is used for estimating probabilities. To ensure that our output is always between 0 and 1 we use sigmoid function. The linear equation having the learned weights and bias is also known as log-odds. The cost function here is called log loss. Regularization becomes very important when it comes to logistic. If no regularisation, then it'll become fully overfit. The model will try to reduce the loss zero (which is not possible in sigmoid), shooting the weights to infinity.

**12. CLASSIFICATION:**

Since the output is probabilistic, we need to have a threshold value to classify the data. In a confusion matrix with all possibilities, A **true positive** is an outcome where the model *correctly* predicts the positive class. Similarly, a **true negative** is an outcome where the model *correctly* predicts the negative class. A **false positive** is an outcome where the model *incorrectly* predicts the positive class. And a **false negative** is an outcome where the model *incorrectly* predicts the negative class. Accuracy is a metric to evaluate our classification model. But it does a poor job when it comes to class imbalance dataset. We have, in additional, Precision and Recall. They should be high for a good ML model.

Feature Crosses (70 min)
Regularization: Simplicity (40 min)
Logistic Regression (20 min)
Classification (90 min)
Video Lecture
Thresholding
True vs. False; Positive vs. Negative
Accuracy
Precision and Recall
Check Your Understanding: Accuracy, Precision, Recall
ROC Curve and AUC
Check Your Understanding: ROC and AUC
Prediction Bias
Programming Exercise
Regularization: Sparsity (20 min)
Neural Networks (65 min)
Training Neural Nets (10 min)
Multi-Class Neural Nets (45 min)
Embeddings (50 min)

## Precision

Explore the options below.

**Consider a classification model that separates email into two categories: "spam" or "not spam." If you raise the classification threshold, what will happen to precision?**

- Definitely increase.
- Probably decrease.
- Definitely decrease.
- Probably increase. ✓

In general, raising the classification threshold reduces false positives, thus raising precision.

Correct answer.



Feature Crosses (70 min)
Regularization: Simplicity (40 min)
Logistic Regression (20 min)
Classification (90 min)
Video Lecture
Thresholding
True vs. False; Positive vs. Negative
Accuracy
Precision and Recall
Check Your Understanding: Accuracy, Precision, Recall
ROC Curve and AUC
Check Your Understanding: ROC and AUC
Prediction Bias
Programming Exercise
Regularization: Sparsity (20 min)
Neural Networks (65 min)
Training Neural Nets (10 min)
Multi-Class Neural Nets (45 min)
Embeddings (50 min)

## Accuracy

Explore the options below.

**In which of the following scenarios would a high accuracy value suggest that the ML model is doing a good job?**

- In the game of roulette, a ball is dropped on a spinning wheel and eventually lands in one of 38 slots. Using visual features (the spin of the ball, the position of the wheel when the ball was dropped, the height of the ball over the wheel), an ML model can predict the slot that the ball will land in with an accuracy of 4%. ✓

This ML model is making predictions far better than chance; a random guess would be correct 1/38 of the time—yielding an accuracy of 2.6%. Although the model's accuracy is "only" 4%, the benefits of success far outweigh the disadvantages of failure.

Correct answer.

- An expensive robotic chicken crosses a very busy road a thousand times per day. An ML model evaluates traffic patterns and predicts when this chicken can safely cross the street with an accuracy of 99.99%.
- A deadly, but curable, medical condition afflicts .01% of the population. An ML model uses symptoms as

Crash Course | Problem Framing | Data Prep | Clustering | Recommendation | Testing and Debugging | GANs

Feature Crosses (70 min)
Regularization: Simplicity (40 min)
Logistic Regression (20 min)
Classification (90 min)
Video Lecture
Thresholding
True vs. False; Positive vs. Negative
Accuracy
Precision and Recall
Check Your Understanding: Accuracy, Precision, Recall
ROC Curve and AUC
Check Your Understanding: ROC and AUC
Prediction Bias
Programming Exercise
Regularization: Sparsity (20 min)
Neural Networks (65 min)
Training Neural Nets (10 min)
Multi-Class Neural Nets (45 min)

## Recall

Explore the options below.

**Consider a classification model that separates email into two categories: "spam" or "not spam." If you raise the classification threshold, what will happen to recall?**

- Always stay constant.
- Always decrease or stay the same. ✓

Raising our classification threshold will cause the number of true positives to decrease or stay the same and will cause the number of false negatives to increase or stay the same. Thus, recall will either stay constant or decrease.

Correct answer.

- Always increase.

https://developers.google.com/machine-learning?authuser=1

Crash Course | Problem Framing | Data Prep | Clustering | Recommendation | Testing and Debugging | GANs

Feature Crosses (70 min)
Regularization: Simplicity (40 min)
Logistic Regression (20 min)
Classification (90 min)
Video Lecture
Thresholding
True vs. False; Positive vs. Negative
Accuracy
Precision and Recall
Check Your Understanding: Accuracy, Precision, Recall
ROC Curve and AUC
Check Your Understanding: ROC and AUC
Prediction Bias
Programming Exercise
Regularization: Sparsity (20 min)
Neural Networks (65 min)
Training Neural Nets (10 min)
Multi-Class Neural Nets (45 min)
Embeddings (50 min)

## Precision and Recall

Explore the options below.

**Consider two models—A and B—that each evaluate the same dataset. Which one of the following statements is true?**

- If model A has better precision and better recall than model B, then model A is probably better. ✓

In general, a model that outperforms another model on both precision and recall is likely the better model. Obviously, we'll need to make sure that comparison is being done at a precision / recall point that is useful in practice for this to be meaningful. For example, suppose our spam detection model needs to have at least 90% precision to be useful and avoid unnecessary false alarms. In this case, comparing one model at {20% precision, 99% recall} to another at {15% precision, 98% recall} is not particularly instructive, as neither model meets the 90% precision requirement. But with that caveat in mind, this is a good way to think about comparing models when using precision and recall.

Correct answer.

- If model A has better recall than model B, then model A is better.