Econometrics

- Assumptions of linear regression
- What is stepwise regression? - Stepwise regression is the step-by-step iterative construction of a regression model that involves the selection of independent variables to be used in a final model. It involves adding or removing potential explanatory variables in succession and testing for statistical significance after each iteration.
- Difference between covariance and correlation?

Covariance	**Correlation**
Covariance is a measure to indicate the extent to which two random variables change in tandem.	Correlation is a measure used to represent how strongly two random variables are related to each other.
Covariance is nothing but a measure of correlation.	Correlation refers to the scaled form of covariance.
Covariance indicates the direction of the linear relationship between variables.	Correlation on the other hand measures both the strength and direction of the linear relationship between two variables.
Covariance can vary between -∞ and +∞	Correlation ranges between -1 and +1
Covariance is affected by the change in scale. If all the values of one variable are multiplied by a constant and all the values of another variable are multiplied, by a similar or different constant, then the covariance is changed.	Correlation is not influenced by the change in scale.
Covariance assumes the units from the product of the units of the two variables.	Correlation is dimensionless, i.e. It's a unit-free measure of the relationship between variables.

- Importance of stationarity?

- What is trend stationary and difference stationarity?
- What is the unit root problem?
- What is a spurious regression? - A “spurious regression” is one in which the time-series variables are non stationary and independent. Or a spurious relationship or spurious correlation is a mathematical relationship in which two or more events or variables are associated but not causally related, due to either coincidence or the presence of a certain third, unseen factor

- How to find if a process is stationary or not?
    - Graphical Analysis - over the period of study the log of GDP has been increasing, that is, showing an upward trend, suggesting perhaps that the mean of the log of GDP has been changing
    - Autocorrelation function and correlogram

- Unit root test - Dickey-Fuller test and Augmented Dickey-Fuller test
- What is specification bias?
- Causes of specification bias - 1. Omission of a relevant variable(s).
  2. Inclusion of an unnecessary variable(s).
  3. Adoption of the wrong functional form.
  4. Errors of measurement.
  5. Incorrect specification of the stochastic error term.
  6. Assumption that the error term is normally distributed

- Consequence of model specification error - (1) underfitting a model, that is, omitting relevant variables, and (2) overfitting a model, that is, including unnecessary variables
- How to check if there is any specification bias? - If there are specification errors, the residuals will exhibit noticeable patterns.
- What are the properties of the best linear unbiased estimator (BLUE)?
- What is the coefficient of correlation (r)? What’s its significance?
- What is the coefficient of determination(r^2)?
- Difference between **regression and correlation**
    - **Correlation Analysis:** Primary objective is to measure degree of linear association between two variables.
    - **Regression Analysis**: We try to estimate or predict the average value of one var on the basis of fixed values of other variables.
- What is the formula of t-value?
- ❖ The width of the confidence interval is proportional to standard error of the estimator,

- What is confidence interval - **Confidence intervals** give us a range of plausible values for some unknown value based on results from a sample
- Strictly speaking a **95% confidence interval means** that if we were to take 100 different samples and compute a **95% confidence interval** for each sample, then

approximately **95** of the 100 **confidence intervals** will contain the true **mean** value (μ)

- Why is the sum of square residuals over the sum of residuals? - By squaring ui hat, this method gives more weight to observations farther from the SRF than those closer to it. Also it avoids the algebraic sum of the ˆui to become zero.
- The term number of **degrees of freedom** means the total number of observations in the sample (= n) less the number of independent (linear) constraints or restrictions put on them.
- What is p-value?

Machine Learning

- difference between outlier and anomaly
  Outlier = legitimate data point that's far away from the mean or median in a distribution.

  Anomaly = illegitimate data point that's generated by a different process than whatever generated the rest of the data

- difference-between-logistic-regression-and-Naive-Bayes
- What-are-the-advantages-of-using-a-decision-tree-for-classification
- svm advantage over logistic and decision tree
- evaluation metrics -False Positive (Type I error), False Negative (Type II error), precision, recall, F1 score, AUC-ROC
- How does a decision tree work?
- How does boosting work? - AdaBoost, GBM and XGBoost
- Recommender system - Content based filtering vs Collaborative filtering

- What are the common regression losses you know about?
  - **Mean Square loss/Quadratic loss/L2 loss -** To calculate the MSE, you take the difference between your model's predictions and the ground truth, square it, and average it out across the whole dataset.

- **Advantage:** The MSE is great for ensuring that our trained model has no outlier predictions with huge errors, since the MSE puts larger weight on theses errors due to the squaring part of the function
- **Disadvantage:** If our model makes a single very bad prediction, the squaring part of the function magnifies the error.

- **Mean Absolute loss/L1 loss** - To calculate the MAE, you take the difference between your model's predictions and the ground truth, apply the absolute value to that difference, and then average it out across the whole dataset
- **Advantage**: The beauty of the MAE is that its advantage directly covers the MSE disadvantage. Since we are taking the absolute value, all of the errors will be weighted on the same linear scale. Thus, unlike the MSE, we won't be putting too much weight on our outliers and our loss function provides a generic and even measure of how well our model is performing
- **Disadvantage:** If we do in fact care about the outlier predictions of our model, then the MAE won't be as effective. The large errors coming from the outliers end up being weighted the exact same as lower errors. This might result in our model being great most of the time, but making a few very poor predictions every-so-often
- **Huber Loss -** Now we know that the MSE is great for learning outliers while the MAE is great for ignoring them. But what about something in the middle?
- The Huber Loss offers the best of both worlds by balancing the MSE and MAE together
- What this equation essentially says is: for loss values less than delta, use the MSE; for loss values greater than delta, use the MAE
- You'll want to use the Huber loss any time you feel that you need a balance between giving outliers some weight, but not too much. For cases where outliers are very important to you, use the MSE! For cases where you don't care at all about the outliers, use the MAE!

- There may be regression problems in which the target value has a spread of values and when predicting a large value, you may not want to punish a model as heavily as mean squared error.
- Instead, you can first calculate the natural logarithm of each of the predicted values, then calculate the mean squared error. **This is called the Mean Squared Logarithmic Error loss, or MSLE for short**.
- It has the effect of relaxing the punishing effect of large differences in large predicted values. As a loss measure, it may be more appropriate when the model is predicting unscaled quantities directly

- What are the common classification losses you know about?
    - Binary Cross Entropy / Log-loss - Cross-entropy will calculate a score that summarizes the average difference between the actual and predicted probability distributions for predicting class 1. The score is minimized and a perfect cross-entropy value is 0.
    - Categorical Cross Entropy - **Softmax** is the only activation function recommended to use with the categorical cross-entropy loss function.

- How does K-means clustering work? KM-clustring vs Hierarchical clustering?

Deep Learning

- Why is ReLU better than sigmoid?
- Advantage of Tanh activation over sigmoid?
- Why don't we use MSE in classification problems?
- How does Gradient Descent work?
- What if forward and backward propagation?
- What is vectorization? - Getting rid of explicit for loops.
- Why do we need non-linear activation functions?

The main steps for building a Neural Network are:

- Define the model structure (such as number of input features and outputs)
- Initialize the model's parameters.
- Loop.
    - Calculate current loss (forward propagation)
    - Calculate current gradient (backward propagation)
    - Update parameters (gradient descent)

- DIfference between parameters and hyperparameters?
    - In summary, model parameters are estimated from data automatically and model hyperparameters are set manually and are used in processes to help estimate model parameters. Model hyperparameters are often referred to as parameters because they are the parts of the machine learning that must be set manually and tuned.
    - Model Parameters: These are the parameters in the model that must be determined using the training data set. These are the fitted parameters.

Hyperparameters: These are adjustable parameters that must be tuned in order to obtain a model with optimal performance.

- Parameters of a standard Neural Network are W and b. Hyperparameters for same are, however, the following:

- Learning rate
- The number of iteration
- The number of hidden layers L
- The number of hidden units n
- Choice of activation functions

- What is the dying RELU problem? How to prevent it?

- Why is random initialization important in NN? Why is it not important in Logistic regression?
    - Logistic Regression doesn't have a hidden layer. If you initialize the weights to zeros, the first example x fed in the logistic regression will output zero but the derivatives of the Logistic Regression depend on the input x (because there's no hidden layer) which is not zero. So at the second iteration, the weights values follow x's distribution and are different from each other if x is not a constant vector
    - If we initialize all the weights with zeros in NN it won't work (initializing bias with zero is ok). If initialized with zeros, then all hidden units would become completely identical (symmetric) and hence compute the exact same function in every iteration. On each gradient descent iteration, all the hidden units will always update in the same way.

- Why initialize weights with small random numbers?
- Why divide data in train, dev and test set? - You will try to build a model upon a training set then try to optimize hyperparameters on the dev set as much as possible. Then after your model is ready you try and evaluate the testing set. Purpose of splitting data into the different category is to avoid overfitting
- What is bias? What is variance?
- How to solve the problem of high Bias? How to solve the problem of high Variance?
- What is regularization ? - Regularization is a technique which makes slight modifications to the learning algorithm such that the model generalizes better. This in

turn improves the model's performance on the unseen data as well. **It helps reduce overfitting**

- What is L1 and L2 regularization ? - These update the general cost function by adding another term known as the regularization term.

  Cost function = Loss (say, binary cross entropy) + Regularization term

  Due to the addition of this regularization term, the values of weight matrices decrease because it assumes that a neural network with smaller weight matrices leads to simpler models. Therefore, it will also reduce overfitting to quite an extent.

- What is weight decay? How is it related with L2 regularization ?
- How does regularization prevent overfitting?
  - Intuition 1 - with lambda
  - If lambda is too large - a lot of w's will be close to zeros which will make the NN simpler (you can think of it as it would behave closer to logistic regression).
  - If lambda is good enough it will just reduce some weights that makes the neural network overfit.

  - Intuition 2 - with tanh activation function
  - If lambda is too large, w's will be small (close to zero) - will use the linear part of the *tanh* activation function, so we will go from non linear activation to *roughly* linear which would make the NN a *roughly* linear classifier.
  - If lambda good enough it will just make some of *tanh* activations *roughly* linear which will prevent overfitting

- What are the different Regularization techniques in Deep Learning?
  - L2 and L1 regularization
  - Dropout
  - Data augmentation
  - Early stopping

- What is dropout regularization?
- Dropout only works during training. We do not use dropout for predicting during testing.
- Why does dropout work?
- What is data augmentation? For example in a computer vision data:
  - You can flip all your pictures horizontally; this will give you more data instances.
  - You could also apply a random position and rotation to an image to get more data.

    - For example in OCR, you can impose random rotations and distortions to digits/letters. New data obtained using this technique isn't as good as the real independent data, but still can be used as a regularization technique.

- What is early stopping? - Early stopping is a kind of cross-validation strategy where we keep one part of the training set as the validation set. When we see that the performance on the validation set is getting worse, we immediately stop the training on the model. This is known as early stopping
- What is exploding and vanishing gradient descent?
- How to solve the problem of vanishing/exploding gradient descent?
    - There is a partial solution that doesn't completely solve this problem but it helps a lot - careful choice of how you initialize the weights - He Initialization / Xavier Initialization
    - Gradient clipping - Checking for and limiting the size of the gradients whilst our model trains is another solution
    - Reduce the complexity of model

- What is batch and mini-batch gradient descent? What is stochastic gradient descent?
    - Mini-batch size:
    - (mini batch size = m) ==> Batch gradient descent
    - (mini batch size = 1) ==> Stochastic gradient descent (SGD)
    - (mini batch size = between 1 and m) ==> Mini-batch gradient descent
    - Batch gradient descent:
    - too long per iteration (epoch)
    - Stochastic gradient descent:
    - too noisy regarding cost minimization (can be reduced by using smaller learning rate)
    - won't ever converge (reach the minimum cost)
    - lose speedup from vectorization
    - Mini-batch gradient descent:
    - faster learning:
        - you have the vectorization advantage
        - make progress without waiting to process the entire training set
    - doesn't always exactly converge (oscillates in a very small region, but you can reduce learning rate)

- Role of exponentially weighted average in deep learning? - The reason why exponentially weighted averages are useful for further optimizing gradient descent algorithms is that they can give different weights to recent data points based on the value of beta. If beta is high (around 0.9), it smoothes out the averages of skewed data

points (oscillations w.r.t. Gradient descent terminology). So this reduces oscillations in gradient descent and hence makes a faster and smoother path towards minima.

- What is gradient descent with momentum? - The momentum algorithm almost always works faster than standard gradient descent. The simple idea is to calculate the exponentially weighted averages for your gradients and then update your weights with the new values.
    - Average on the vertical direction will be close to 0 - smaller oscillation help to take straightforward path
    - Average on the horizontal direction will be big
    - Momentum helps the cost function to go to the minimum point in a more fast and consistent way.
    - beta is another hyperparameter. beta = 0.9 is very common and works very well in most cases.

- What is RMSProp (Root mean square propagation) ? - copy
- What is Adam optimization algorithm? - Adaptive Moment Estimation
    - Adam optimization simply puts RMSprop and momentum together

- What is learning rate decay and why do we use it?
- What is the problem of local optima? - The normal local optima is not likely to appear in a deep neural network because data is usually high dimensional. For point to be a local optima it has to be a local optima for each of the dimensions which is highly unlikely
- What is batch normalization? - To increase the stability of a neural network, batch normalization normalizes the output of a previous activation layer by subtracting the batch mean and dividing by the batch standard deviation.
    - It reduces overfitting because it has a slight regularization effect. Similar to dropout, it adds some noise to each hidden layer's activations. Therefore, if we use batch normalization, we will use less dropout, which is a good thing because we are not going to lose a lot of information.

- What is softmax regression ? - There are a generalization of logistic regression called Softmax regression that is used for multiclass classification/regression

❖CNN

- What is the difference between Convolution layers and normal dense layers?
    - The main functional difference of convolution neural network is that, the main image matrix is reduced to a matrix of lower dimension in the first layer itself through an operation called Convolution
    - Networks having large number of parameter face several problems, for e.g. slower training time, chances of overfitting

- Early layers of CNN might detect edges then the middle layers will detect parts of objects and the later layers will put the these parts together to produce an output
- What is padding and why do we use it?
    - The convolution operation shrinks the matrix. we saw that a 6x6 matrix convolved with 3x3 filter/kernel gives us a 4x4 matrix
    - We want to apply convolution operation multiple times, but if the image shrinks we will lose a lot of data on this process. Also the edges pixels are used less than other pixels in an image - So the problems with convolutions are:
    - Shrinks output.
    - throwing away a lot of information that is on the edges.

- To solve these problems we can pad the input image before convolution by adding some rows and columns to it. We will call the padding amount P the number of row/columns that we will insert in top, bottom, left and right of the image

- What is strided convolution? - When we are making the convolution operation we used S to tell us the number of pixels we will jump when we are convolving the filter/kernel.

- no matter the size of the input, the number of the parameters is the same if filter size is the same. That makes it less prone to overfitting.

- What are pooling layers - pooling layers to reduce the size of the inputs, speed up computation, and to make some of the features it detects more robust.
    - The max pooling is saying, if the feature is detected anywhere in this filter then keep a high number. But the main reason why people are using pooling because its works well in practice and reduce computations
    - Max pooling has no parameters to learn
    - Average pooling is taking the averages of the values instead of taking the max values

Hyperparameters summary

- f : filter size.

- s : stride.
- Padding are rarely uses here.
- Max or average pooling.

Here are some classical CNN networks:
- LeNet-5
- AlexNet
- VGG

Some data augmentation methods that are used for computer vision tasks includes:
- Mirroring.
- Random cropping.
    - The issue with this technique is that you might take a wrong crop.
    - The solution is to make your crops big enough.
- Rotation.
- Color shifting

- What is Object detection? - Given an image we want to detect all the objects in the image that belong to a specific class and give their location. An image can contain more than one object with different classes.
- What is object localization? - Given an image we want to learn the class of the image and where the class location is in the image. We need to detect a class and a rectangle of where that object is.

- To make classification with localization we use a Conv Net with a softmax attached to the end of it and a four numbers bx, by, bh, and bw to tell you the location of the class in the image. The dataset should contain this four numbers with the class too.

- What is landmark detection? - In some of the computer vision problems you will need to output some points. That is called landmark detection.
    - For example, if you are working in a face recognition problem you might want some points on the face like corners of the eyes, corners of the mouth, and corners of the nose and so on. This can help in a lot of applications like detecting the pose of the face.

- How does the sliding window algorithm work in object detection?

Sliding windows detection algorithm:

i. Decide a rectangle size.
ii. Split your image into rectangles of the size you picked. Each region should be covered. You can use some strides.
iii. For each rectangle feed the image into the Conv net and decide if its a car or not.
iv. Pick larger/smaller rectangles and repeat the process from 2 to 3.
v. Store the rectangles that contain the cars.
vi. If two or more rectangles intersects choose the rectangle with the best accuracy.

- Disadvantage of a sliding window is the computation time.

- The weakness of the algorithm is that the position of the rectangle won't be so accurate. Maybe none of the rectangles is exactly on the object you want to recognize.

- What is the YOLO algorithm and Bounding Box prediction?
- What is Intersection over Union (IOU) ?
- What is Non-Max Suppression ?
- What is the use of Anchor boxes ?- If there are more than one object in one bounding box then we use the concept of anchor boxes.
    - If there are two anchor boxes and three object in a bounding box then this algo doesn't work for this case

❖RNN

- How does an RNN work?
- RNN vs basic neural network?
- LSTM vs basic RNN?
- What are the gates present in LSTM and their functions?
- What is an attention layer?
- What is an embedding layer? What does a 50-dimension embedding vector mean?
- Hyperparameters in LSTM?