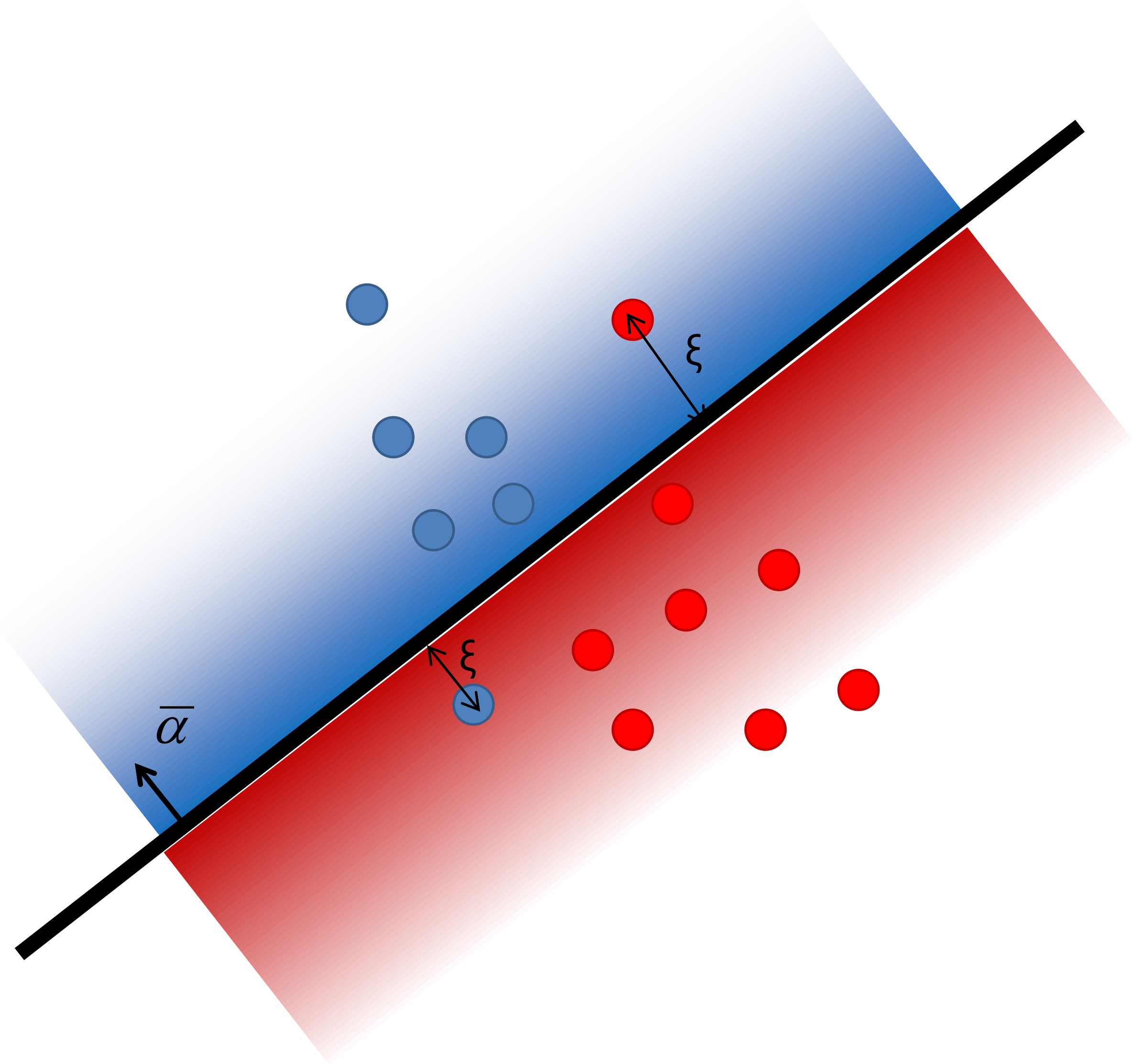
ML Interview



Dmitry Ryabokon, github.com/dryabokon

Part 1: General approach

1. Trade-off between bias and variance
2. Methods to avoid an overfit
3. Reducing data dimensions
4. Rotation in PCA
5. # of variables > # of observation
6. Imbalanced data set
7. Stratified sampling
8. Multicollinearity
9. Correlation: continuous & categorical
10. Cross validation for time series
11. Accuracy, precision, recall, F1 score
12. Missing values: the approach
13. Probability and likelihood
14. Variable importance chart
15. Variance inflation factor VIF

https://habr.com/ru/company/ods/blog/325654/

Проклятие размерности

Back propagation

Feed forward

Regression: linear, log

About RF

Part2: The models

21. Regressions: LM, log, etc
22. Penalized regression models
23. KNN vs K-means
24. Decision tree, how is pruned?
25. RF vs GBM
26. Ensambling
27. Reinforcement learning
28. GAN

https://habr.com/ru/company/ods/blog/322534/

Part 3: tricky questions

22. Backpropagation
24. Big data tools for machine learning: Spark

include sorting (plus searching and binary search),
divide-and-conquer,
dynamic programming/memoization,
greediness,
recursion or algorithms linked to a specific data structure.
Know Big-O notations (e.g. run time) and
be ready to discuss complex algorithms like Dijkstra and A*

Part 4: geeks4geeks link

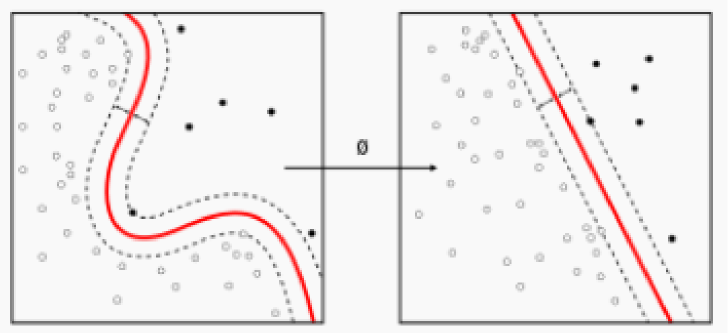
Part 1

General approach

1. Trade-off between bias and variance

The **bias** is an error from wrong assumptions in the learning algorithm
High bias can cause **underfit**: the algorithm can miss relations between features and target

The **variance** is an error from sensitivity to small fluctuations in the training set.
High variance can cause **overfit**: the algorithm can model the random noise in the training data, rather than the intended outputs



2. Methods to avoid an overfit (high variance problem)

1- Keep the model simpler: reduce variance by taking into account fewer variables and parameters, thereby removing some of the noise in the training data.
2- Use cross-validation techniques such as k-folds cross-validation.
3- Use regularization techniques such as LASSO that penalize certain model parameters if they're likely to cause overfitting. Higher model coefficients get penalized, hence lowering model complexity.
4- Use top n features from variable importance chart. May be, with all the variable in the data set, the algorithm is having difficulty in finding the meaningful signal.

3. Reducing data dimensions

1. Separate the numerical and categorical variables and remove the correlated variables.
For numerical variables, we'll use correlation. For categorical variables, we'll use chi-square test.

2. Using online learning algorithms like Vowpal Wabbit (available in Python) is a possible option.
3. Building a linear model using Stochastic Gradient Descent is also helpful.

5. When # of variables > # of observation

In such high dimensional data sets, we can’t use classical regression techniques, since their assumptions tend to fail. When p > n, we can no longer calculate a unique least square coefficient estimate, the variances become infinite, so OLS (Ordinary Least Squares) cannot be used at all.

To combat this situation, we can use penalized regression methods like lasso, LARS, ridge which can shrink the coefficients to reduce variance. Precisely, ridge regression works best in situations where the least square estimates have higher variance.

6. Imballanced dataset

1- Collect more data to balance the dataset.
2- Resample the dataset
3- Try a different algorithm altogether on your dataset.

8. Multicollinearity

Multicollinearity is problem that you can run into when you’re fitting a regression model, or other linear model. It refers to predictors that are correlated with other predictors in the model. Unfortunately, the effects of multicollinearity can feel murky and intangible, which makes it unclear whether it’s important to fix.

Moderate multicollinearity may not be problematic. However, severe multicollinearity is a problem because it can increase the variance of the coefficient estimates and make the estimates very sensitive to minor changes in the model. The result is that the coefficient estimates are unstable and difficult to interpret. Multicollinearity saps the statistical power of the analysis, can cause the coefficients to switch signs, and makes it more difficult to specify the correct model

Upon correlation matrix - **remove high correlated variables** (eg above 75%).

In addition, calculate **VIF (variance inflation factor) to check the presence of multicollinearity**.
VIF value <= 4 suggests no multicollinearity whereas a value of >= 10 implies serious multicollinearity.

9. Correlation between continuous and categorical variable

ANCOVA

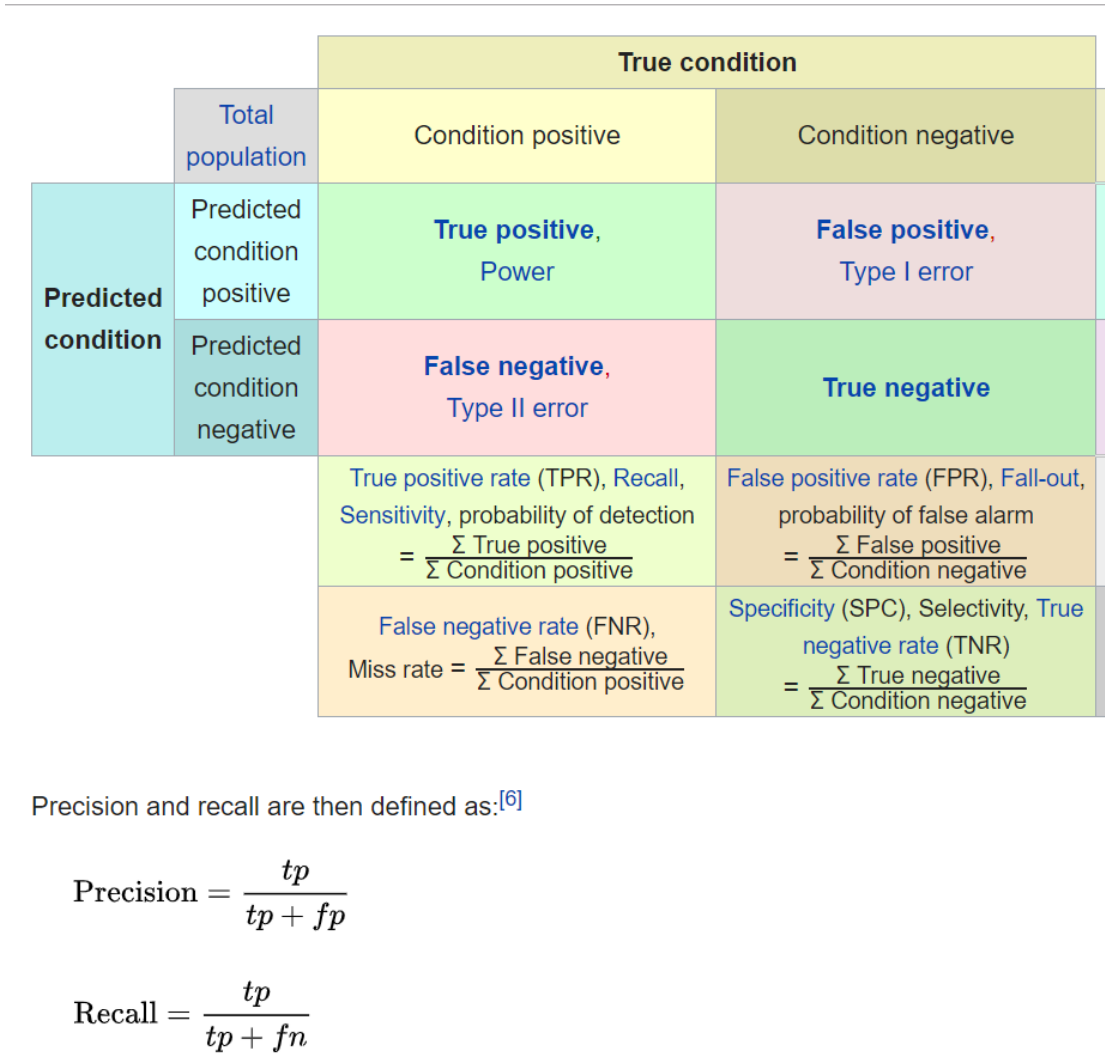
11. Model performance: Accuracy, Precision, Recall, F1

Accuracy = It is the number of correct predictions made divided by the total number of predictions made

Precision = number of positive predictions divided by the total number of positive class values predicted.

Recall = number of positive predictions divided by the number of positive class values in the test data.

F1 Score = 2*((precision*recall)/(precision+recall))



		True condition	
	Total population	Condition positive	Condition negative
Predicted condition	Predicted condition positive	**True positive**, Power	**False positive**, Type I error
	Predicted condition negative	**False negative**, Type II error	**True negative**
		True positive rate (TPR), Recall, Sensitivity, probability of detection $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$
		False negative rate (FNR), Miss rate $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$	Specificity (SPC), Selectivity, True negative rate (TNR) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$

Precision and recall are then defined as:[6]

$$\text{Precision} = \frac{tp}{tp + fp}$$

$$\text{Recall} = \frac{tp}{tp + fn}$$

13. Probability and likelihood

Likelihood is the probability that an event that has already occurred would yield a specific outcome.

Probability refers to the occurrence of future events, while a likelihood refers to past events with known outcomes.

14. Variable importance

Following are the methods of variable selection you can use:

- Remove the correlated variables prior to selecting important variables
- Use linear regression and select variables based on p values
- Use Forward Selection, Backward Selection, Stepwise Selection
- Use Random Forest, Xgboost and plot variable importance chart
- Use Lasso Regression
- Measure information gain for the available set of features and select top n features accordingly.

Part 2

The models

22. Penalized regression models: ridge (L2) and lasso (L1)

Regularization becomes necessary when the model begins to ovefit / underfit.

This technique introduces a cost term for bringing in more features with the objective function.

Hence, it tries to push the coefficients for many variables to zero and hence reduce cost term. This helps to reduce model complexity so that the model can become better at predicting (generalizing).

The **key difference** between these techniques is that Lasso shrinks the less important feature’s coefficient to zero thus, removing some feature altogether. So, this works well for **feature selection** in case we have a huge number of features.
Traditional methods like cross-validation, stepwise regression to handle overfitting and perform feature selection work well with a small set of features but these techniques are a great alternative when we are dealing with a large set of features.

In presence of many variables with small / medium sized effect, use ridge regression.
Conceptually, we can say, lasso regression (L1) does both variable selection and parameter shrinkage, whereas Ridge regression only does parameter shrinkage and end up including all the coefficients in the model. In presence of correlated variables, ridge regression might be the preferred choice. Also, ridge regression works best in situations where the least square estimates have higher variance.

23. How KNN is different from K-Means

KNN is a classification (or regression) algorithm
It tries to classify an unlabeled observation based on its k (can be any number) surrounding neighbors. It is also known as lazy learner because it involves minimal training of model.

K-means is an unsupervised clustering algorithm

24. Decision tree algorithm

You've built a random forest model with 10000 trees. You got delighted after getting training error as 0.00. But, the validation error is 34.23. What is going on? Haven't you trained your model perfectly?
Answer: The model has overfitted. Training error 0.00 means the classifier has mimiced the training data patterns to an extent, that they are not available in the unseen data. Hence, when this classifier was run on unseen sample, it couldn't find those patterns and returned prediction with higher error. In random forest, it happens when we use larger number of trees than necessary. Hence, to avoid these situation, we should tune number of trees using cross validation.

25. RF vs GBM

RF uses bagging technique to make predictions.
GBM uses boosting techniques to make predictions.

In bagging technique, a data set is divided into n samples using randomized sampling. Then, using a single learning algorithm a model is build on all samples. Later, the resultant predictions are combined using voting or averaging. Bagging is done is parallel.

In boosting, after the first round of predictions, the algorithm weighs misclassified predictions higher, such that they can be corrected in the succeeding round. This sequential process of giving higher weights to misclassified predictions continue until a stopping criterion is reached.

Random forest improves model accuracy by reducing variance (mainly). The trees grown are uncorrelated to maximize the decrease in variance. On the other hand, GBM improves accuracy my reducing both bias and variance in a model.

References

https://www.analyticsvidhya.com/blog/2016/09/40-interview-questions-asked-at-startups-in-machine-learning-data-science/

https://www.springboard.com/blog/machine-learning-interview-questions/

https://www.edureka.co/blog/interview-questions/python-interview-questions/

https://www.udemy.com/cracking-python-interview-questions-on-programming-12-hrs/