# Data Analytics

Bryan Hoang

2021-12-01

# Contents

# 1 Lecture 6

2021-10-20

## 1.1 Introduction

- Using basketball dataset (win loss & margins) found on OnQ under week 6.
- What's the point of these datasets?
  - The use of a model made form this data depends from case to case. There's no single rule for the usefulness of a model.
  - Hindsight?

## 1.2 Design = Good Choices

Figure 1.1: Design diagram

- Generic workflow: file read → binning & normalization & missing value inputation → model → scorer
  - Diagram (Based James?)
  - Random forests and Bagging/boosting diverge from the more typical workflows.
- When would one use binning?
  - Mapping numerical values to categorical values which better reflect meaning (e.g., numeric ages to categorical age ranges)

- When would one use normalization?
  - Depends on what will happen later in the workflow.
  - e.g., using neural networks, k-nearest neighbour
- Should try to design from "Right to left" of workflow since we're usually focused on the end result.
- Next stage:
  - Partitioning
  - cross-validation (e.g., x-partition, x-aggregator)
  - Bootstrap sample or Out Of Bag (OOB)
- Then we build the model
- Finally, score the results
- Clustering is thrown into all of that somehow
- Types of Models:
  - One R (Never use)
  - Bayes
    - Bad for redundant attributes
    - Good when there are many attributes that may be useful
  - k-NN
    - Bad for datasets with a large number of datapoints (big n), since scales based off of the size of the dataset
    - Great for datasets where the classes aren't clumped together (geometrically). Hard to determine with testing it.
  - Decision trees (Never use)
  - SVMs
    - Bad for problems with a large number of classes
    - Pretty good go to option otherwise (always)
  - Rules (Never use)
  - Random forests
    - No immediate major downsides
    - Therefore, practically always a good option
  - Neural networks
    - Bad if training cost is an important factor (big n)
    - Great when there's a small amount of info in each attribute. e.g., images, audio
- From the left side, use knowledge of dataset to make decisions on the workflow
- From the right side, use knowledge of the problem to make decisions on the workflow

# 2 Lecture 7

2021-10-26

- Looking at basketballwinloss data
- RF out of bag sampling
  - Looking at prediction confidences can help determine why model might be misbehaving.
  - Attribute statistics help see what the RF is determining to be a significant attribute.
- Tree Ensemble learner is a generalization of RF (not the same thing, doesn't select attributes).
  - Options: Tree Options & Ensemble configuration.
  - Out of bag predictions: model count should avg about a third
- A method for determining the most important attributes involves using a math function & sorting the important (splits times candidates)
  - Iteratively remove unimportant attributes to try and increase prediction accuracy.

# 3 Lecture 8

2021-11-3

- Used Z-scoring method of Normalizer node to improve performance of Tree Ensemble Learner.
- Explaining why clustering is useful.
- Using k-Means clustering in KNIME, analyzing how wins/losses are clustered (ain't prefect).
- Certain parts of Scatter matrix help analyze effectiveness of clusters (include win/loss and cluster columns).
- Scatter plot with high jitter in appearance also helps

# 4 Lecture 9

2021-11-10

- Using provided Matlab script on data to make visual graphs
-

# 5 Lecture 10

2021-11-17

The man has something against Nancy Pelosi?

## 5.1 Distance Based Clustering

- Discussing distance based clustering (and going through KNIME workflow)
- Configuration:
  - Different distance calculations. Can't go wrong with Euclidean. Also have Hamming and $L^\infty$.
  - DBSCAN — $\varepsilon$ a challenge to configure, as it depends more on local properties of the dataset and less on the global properties. Should play around by decreasing it to increase the # of clusters.
  - DBSCAN — Min # of clusters can sometimes be ignored, depending on initial results.
- Also pretty bad for the basketball data set
- Recall last week that trying SVD on a very "blobbly" dataset didn't result in good predictions.

## 5.2 Visualization

- Chernoff faces
  - Very easy for humans to see what face is an outlier depending on which attribute we tend to zero in on as "different".
  - Things get weird when you try to use Chernoff faces on Tinder to find "the one". The faces may make you biased towards someone who wasn't what you were originally looking for
  - Baseball Chernoff Faces
  - . . . Are they useful? Nahhh. Funny nonetheless.
- Back to Knime:
  - Visualizing correlation matrices. e.g., Lack of red (negative correlation) may indicate a bias towards collecting data only on indicators of winning, not on indicators of losing.
  - Using a Scatter plot to visualize relationships between attributes of interest. Avoid binary attributes if possible since they don't reveal too much information of there's lots of overlap.
  - Try using a Histogram as well! Customize attributes and binning to ascertain properties such as distribution.

  - Head Map: Coloured Planar view of the dataset. Answer Q's like "Are most winning games the same?", see some differences between winning and losing heat maps
  - Parallel coordinate plot: Hard to explain with words, pretty picture (kinda looks like a Neural net where a layer is instead an axis for an attribute, and the paths are records now). Helps view correlation based on amount of crossover of records between parallel attribute lines. Highlighting different records can reveal insights such as "losing games can still have high free throw stats compared to highlighted winning games". Can rearrange attribute axis, for reasons. Doesn't scale to large # of attributes, just like for Neural networks! Hmmm.

# 6 Lecture 11: Social media and data analysis

2021-11-24

Big Tech companies, like Facebook & Google, tend to not use Data Analytics in the most ethical way...

## 6.1 Danger #1: Selling your *Personally Identifiable Information (PII)* & attention for free stuff

- Big Tech gives you personalize ads and free stuff
- You give them PII and attention
- Repeat the cycle initiated by the Big Tech company ("stickiness"). e.g., Misinformation i prevalent since people may be seeking it out.
- Analogy: You are the "*Dragon*"
- The D.A. they use improves personalization, and eventually narrow down on **price sensitivity**. Then they sell it to everyone else

## 6.2 Danger #2: Social media is inherently biased against you

- People tend to post good things, not bad things
- The structure reinforces the bias
- e.g., The Friend Paradox - Homophily (ho-moph-ily),"birds of a feather flock together".

  Insert Figure here.

## 6.3 Previous Exam Question

- Q's will have a "twist", to make us think about what the actual problem to solve is. Don't just apply the 5 steps of data analytics.
- e.g., Baseball data sorted based on ascending salary (400 k - 2.8 mil). How does a player's performance relate to their salary, and whether it's fair? (won't be given data on exam, mostly for demonstration during lecture)
  - To assess "fairness", consider **clustering** players to see if salaries do reflect similarly performant players. Be careful with ranking/selecting attributes.
  - Some figuress from SVD is shown in MATLAB, although we won't have this on the exam.
  - Seeing high performing players with low salaries amongst other high paid players implies players are being paid unfairly.
  - In KNIME, looking at correlation between attributes for analyzing attributes. Then used EM clustering (good go to). To decide # of clusters, try a couple (3-5). Meh results, as expected form SVD figures form earlier.
  - Few data points (e.g., potential outliers) should be taken with a grain of salt,
  - Conclusion: It is unfair. How to fix it? Based on analysis.

# 7 Lecture 12: Design Questions

2021-12-01

## Customer Price Sensitivity

- Online retailers, such as Amazon, has the advantage of adjusting "shelf" prices easily to determine people's individual price (in)sensitivities.
  - Columns: 2000 most popular products
  - Rows: Customers
  - Entries:
    - 0: Customer hasn't bought.
    - 1: Customer bought at **lowest** price.
    - 2: " **midranged** ".
    - 3: " **highest** ".
- How to predict price insensitivity?
- Cluster so that price insensitive people are farther away from the origin. (e.g., 3s vs 0s).
- To make prediction easier, consider designing a label. How? Row sum divided by # of non-zero entries.
- What type of clustering?
  - K-means? Nah, groups seem more like onion slices rather than blobs
  - EM? Reasoning for clusters makes sense. e.g., frugal, middle, cash money. So should expect 3 reasonable groups
    - Spoiler: It does!
  - Density? Nah. Nothing alludes to density based clusters.
  - Hierarchical? Too many rows (large amount of data), and we don't care about the fine details.
  - SVD? Maybe. 2k dimensional space into 3d space.
    - Sometimes doesn't reveal clusters with clean separation.
- Prediction (based on new label):
  - Baye's Rule? Too many attributes.
  - kNN? Might be good based on EM clustering results.
  - NN? Nope, a lot of attributes and data and attributes may not be related to each other.
  - SVM? Meh.
  - RF? Probably!
    - Pretty accurate!

## Outline for an answer

- An answer. We can design a way to calculate the price insensitivity by first designing a label (details). One digression has an issue (mention it). Another digression, all products aren't created equal. So we'll weight the score function based on the average of a attribute and the std dev of the column sums. Then talk about **clustering** to help decide on a predictor and analyze other properties of the dataset. Argue for appropriate clustering methods and why others aren't reasonable. e.g., ones that are clearly good or clearly bad. Then talk about predictors in a similar manner. Then talk about actionable tasks from design. Continue to mention digressions to aim for that A+.

- Comment on weaknesses (i.e., digressions) to suggested approaches and think of alternatives/further improvements to get an A+.