Data Science for Business
Lecture #10
Review of Data Science for Business

Prof. Alan L. Montgomery

The University of Hong Kong & Carnegie Mellon University, Tepper School of Business

email: alanmontgomery@cmu.edu

Outline

Building an Analytics Checklist

Course Review

Modeling Review

Final Thoughts

Building an Analytics Checklist

What were the pitfalls and remedies?

What are pitfalls for each analysis?

Prescriptive Analytics

- Narrow focus on single goal when problem is multi-faceted
- Inaccurate data makes the optimization meaningless

Predictive Analytics

- The model overfits the training data and does not generalize
- Lack of ability to explain and interpret the model

Descriptive Analytics

- Availability Bias: Is the data that we have what we want?
- Narrative Bias: Myopic within the parameters of the model

Checklist: Prescriptive Models

Metrics. Does the problem have multiple-objectives? How can competing objectives be incorporated in the formulation?

Calibration. Has the model been calibrated with existing data to yield reasonable outputs?

Integer variables. Have integer variables been used sparingly (preferably only for binary choices)? Will the proposed solution methods scale?

Alternate solutions. Can alternate solutions that are nearly as good be generated? Can a search for finding them be tailored to satisfy some 'diplomatic' requirements?

Binding constraints. What important resource constraints are binding at the optimum? Can economic shadow prices be imputed for them?

Simulation Distributions. Is the set of scenarios generated in the simulation sufficiently representative of the situations to which the model will be exposed?

Checklist: Predictive Models

Sampling. Has the sample been collected in a representative way?

Data spending. Was there enough data to carve out a validation sample for comparing models in addition to the training and test sets?

Decision Tree Relationships. Can you demonstrate the relationships and interactions reflected in the tree directly from the data (using visualizations or other statistics)?

Logistic Regression Variable Importance. Have the significant variables been ordered in rank of importance based on their impact on the odds ratio of the outcome? (Alternately, were the variables standardized before running the regression?)

Regression Variable Selection. Have highly correlated predictors been included in the analysis? If so, can a smaller representative set be selected?

Model Complexity. Is there a much simpler and explainable model that gives quality metrics that are only slightly worse than the best possible complex model for this problem?

Sanity check. Do the relationships suggested in the models make sound business sense? (E.g. customer id versus time of joining)

Correlation versus Causation. Do we believe that if an input is changed will it cause a change in the output?

Checklist: Descriptive Models

Variable scales. Have variables over different ranges been scaled to give each of them the same (or the intended) impact in the clustering?

Clustering Variable Selection. Are variables that represent the same features repeated in the clustering exercise to lead to overweighting these features?

Significance of Patterns. Are rules or patterns found by descriptive methods supported by sufficient proportion of the data (support) and with sufficiently high discrimination (confidence and lift)?

Visualizations that enable interpretation. Are there multiple visualizations that support any of the interesting patterns discovered?

Stationarity. Is the data that you have collected stable (e.g. over time, geography, demographic groups...)?

Sampling biases. Has any of the data used already been modified/filtered/censored by business rules applied?

Risks

Privacy
- Creep factor
- Trade-off between sensitivity and effectiveness

Security
- Sensitive data can get hacked
- Governance and access violations
- Confidentiality and cross-unit walls

Legality
- Cross-border stipulations (e.g. “safe harbor”)
- Fusion of data sources
- PII concerns in many sectors: health, education, public services

Ethicality
- Effects of automation on workforce and culture
- Discrimination due to model recommendations

Key Takeaways
Pitfalls on the way to impact

Data. Data is hard to get due to availability, architecture and legacy issues. It is harder to get due to human issues such as the lack of information or control that managers perceive in functional organizations that operate in silos

Model. Models may not address the appropriate objective. Even when they do, they may not choose the relevant features or predictors, may not be valid enough to be generalizable, or be interpretable to explain to a client, or be flexible to fit a broader set of new situations.

Human factors. Beware of human managers that can wreck a careful data study in the form of hippos, white elephants and holy cows!

Risks with Data Analytics. Be mindful of three key risks that always exist in dealing with valuable data: security, privacy and legality

Course Review

Course Objectives

Understand the data mining process

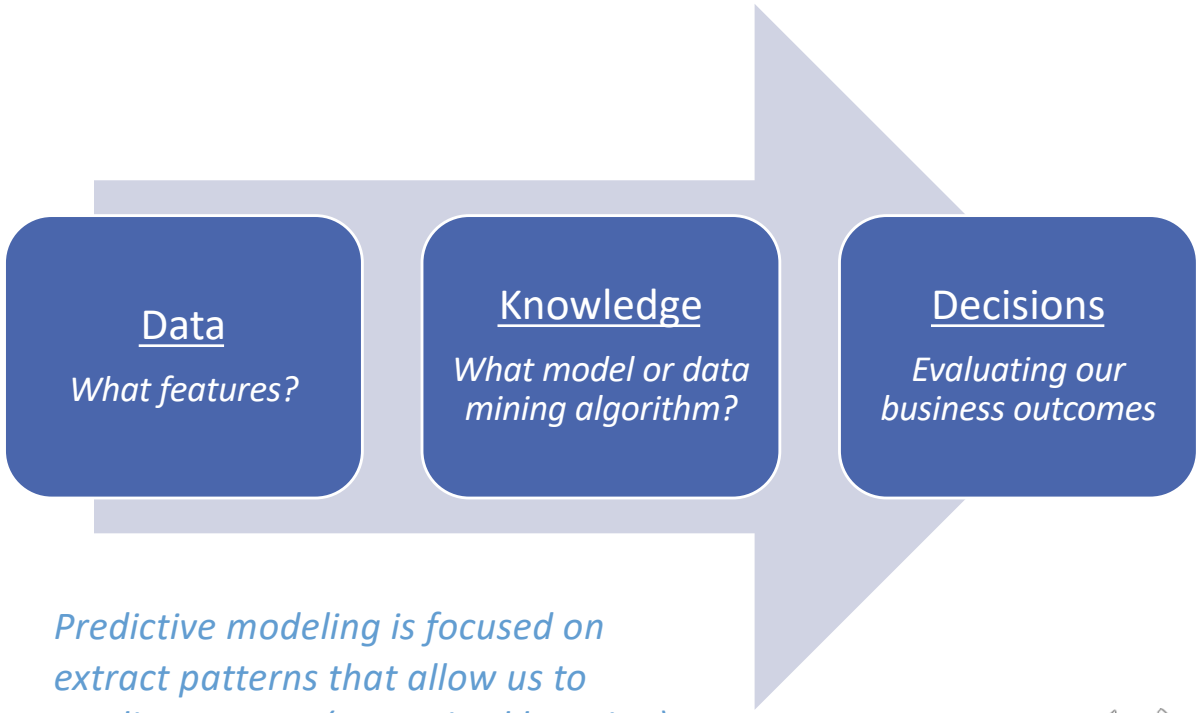
Introduce and apply a set of data mining tools

Apply these techniques to specific case studies to solve business and e-commerce problems

What is Data Science?

Extracting *knowledge* from *data* to make better *decisions* in business.

- Data can be structured (transactions) or unstructured (emails, videos, photos, social media, …).
- Data are facts, while knowledge is generalizable.
- The extraction process uses data mining techniques.
- Data scientists need to have both technical skills for data mining, but domain knowledge to know how to structure an analysis to have an impact.
- Notice that Data Science is an empirical science. It is based upon experimentation or observation (evidence), as opposed to case studies or theoretical research methods.



Predictive modeling is focused on extract patterns that allow us to predict a target (supervised learning)

What is “Data”?

Data is a set of values of qualitative or quantitative variables, or more simply it is information.

Data is measured, collected, and analyzed through reports, models or visualizations

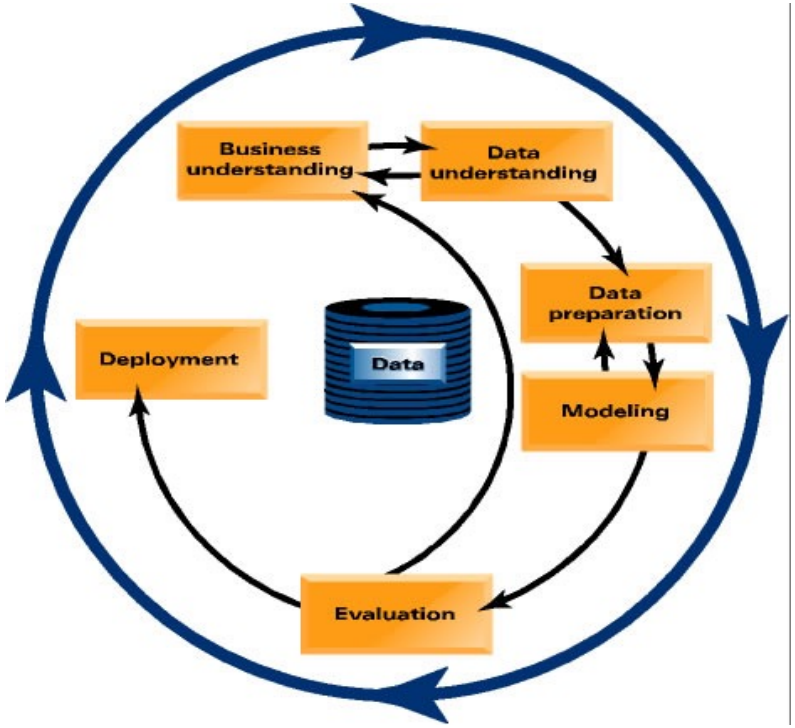
What are its strengths and weaknesses of an empirical science?

Strength: Objectivity, measurability, replicability, …

Weaknesses: Inefficient, Potential for bias in data due to (bad) prior research or incorrect collection, Lack of information (e.g., predict future), …

What is “Science” part?

Finding Patterns using Data Mining Process



Source: Foster Provost

Predictive Models

Use a set of *inputs* (or features|covariates|attributes|independent variables) to forecast an *output* (or target|dependent)

It is an *abstraction*. Abstractions are meant to capture the most important elements – they are not perfect.

They are *mathematical* models. We quantify the relationships between the inputs and outputs

They are not perfect, so usually we think in *probabilistic* terms and use *statistical* techniques

Optimal Pricing

- *Inputs*: Vector of all prices and past purchases
- *Output:* Category profit
- *Model:* Log-linear regression

Customer Churn

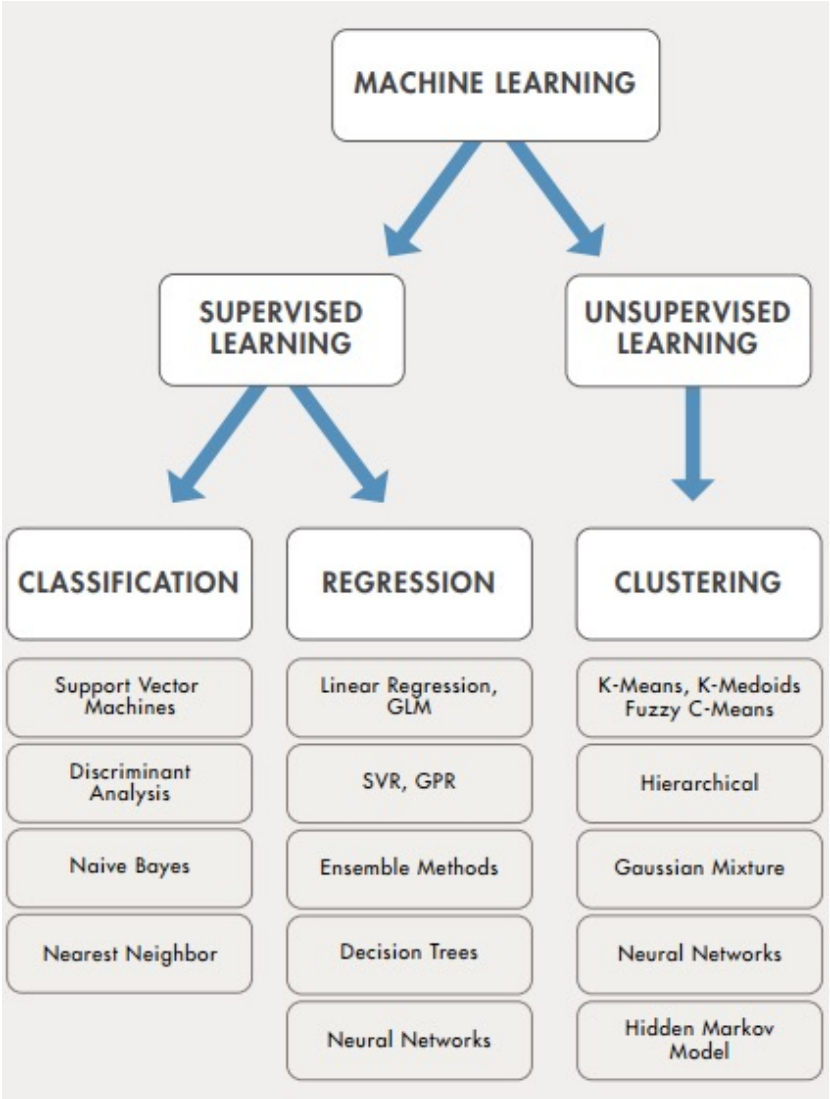
- *Inputs:* Vector of customer characteristics (tenure, usage, …)
- *Output:* Probability of churn next month
- *Model:* Logistic regression

Production Recommendation

- *Inputs:* Vector of all past purchases for a customer
- *Output:* Vector of purchase probabilities for all new products
- *Model:* k-nearest neighbor

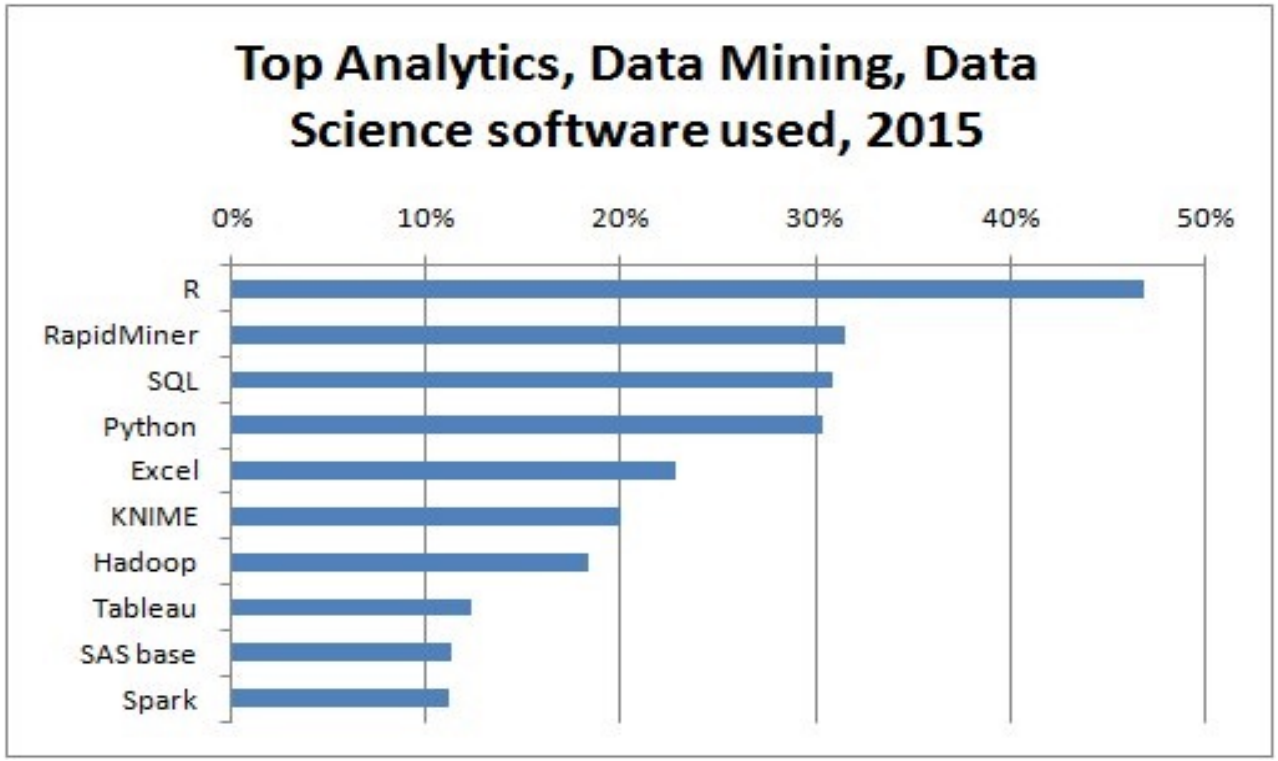
Ten popular Data Mining Algorithms

1. Linear regression
2. Logistic Regression
3. K-Means Clustering
4. K-Nearest Neighbors (KNN) Classification
5. Naive Bayes Classification
6. Decison Trees
7. Support Vector Machine (SVM)
8. Artifical Neural Network (ANN)
9. Apriori
10. AdaBoost



https://www.datasciencecentral.com/profiles/blogs/machine-learning-summarized-in-one-picture

What type of software?



R Commands

- **Linear regression**: "lm" method from base package could be used for linear regression models. Following is the sample command:

```
lm_model <- lm(y ~ x1 + x2, data=as.data.frame(cbind(y,x1,x2)))
```

- **Logistic Regression**: Logistic regression is a classification based model. "glm" method from base R package could be used for logistic regression. Following is the sample command:

```
glm_model <- glm(y ~ x1+x2, family=binomial(link="logit"), data=as.data.frame(cbind(y,x1,x2))
```

- **K-Means Clustering**: "kmeans" method from base R package could be used to run k-means clustering. Following is a sample command given X is a data matrix and m is the number of clusters:

```
kmeans_model <- kmeans(x=X, centers=m)
```

- **K-Nearest Neighbors (KNN) Classification**: "knn" method from "class" package could be used for K-NN modeling. One need to install and load "class" package. Following is the sample command given X_train represents a training dataset, X_test represents test data set, k represents number of nearest neighbors to be included for the modeling

```
knn_model <- knn(train=X_train, test=X_test, cl=as.factor(labels), k=K)
```

- **Naive Bayes Classification**: "naiveBayes" method from "e1071" package could be used for Naive Bayes classification. One need to install and load "e1071" package prior to analysis. Following is the sample command:

```
naiveBayes_model <- naiveBayes(y ~ x1 + x2, data=as.data.frame(cbind(y,x1,x2)))
```

- **Decision Trees**: "rpart" method from "rpart" can be used for Decision Trees. One need to install and load "rpart" package. Following is the sample command:

```
cart_model <- rpart(y ~ x1 + x2, data=as.data.frame(cbind(y,x1,x2)), method="class")
```

- **Support Vector Machine (SVM)**: "svm" method from "e1071" package could be used for SVM. Note that the same package also provide method, naiveBayes, for Naive Bayes classification. One need to install and load "e1071" package. Following is the sample command given X is the matrix of features, labels be the vector of 0-1 class labels, and C being regularization parameter

```
svm_model <- svm(x=X, y=as.factor(labels), kernel ="radial", cost=C)
```

- **Artifical Neural Network (ANN)**: "neuralnet" method from "neuralnet" package could be used for ANN modeling. Following is sample command:

```
ann_model <- neuralnet( y ~ x1 + x2 + x3, data=as.data.frame(cbind(y,x1,x2, x3)), hidden = 1)
```

Prediction could be made using following formula:

```
p <- compute( ann_model, as.data.frame(cbind(x1,x2)) )
```

- **Apriori**: "apriori" method from "arules" package could be used for Apriori analysis. One need to install and load "arules" package. Following is the sample command:

```
apriori_model <- apriori(as.matrix(sampleDataset), parameter = list(supp = 0.8, conf = 0.9))
```

- **AdaBoost**: "ada" method from "rpart" package could be used as boosting function. Following is sample command:

```
boost_model <- ada(x=X, y=labels)
```

For most of the above formulas including linear regression model, one could use following function to predict:

```
predicted_values <- predict(some_model, newdata=as.data.frame(cbind(x1_test, x2_test)))
```

http://vitalflux.com/cheat-sheet-10-machine-learning-algorithms-r-commands/

Modeling Review

Logistic Regression Example: *Mathematical Representation*

What is the relationship between the output (or probability of an event occurring) and the inputs (or features)?

Mathematical formula:

$$\Pr(y_i = 1) = \frac{\exp\{z_i\}}{1+\exp\{z_i\}} = \frac{1}{1+\exp\{-z_i\}},$$

$$\text{where } z_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_p x_{pi}$$

If we want to predict sales in a retail store what covariates should we use?

Logistic Regression Example: *Graphical Intuition*

What is the relationship between the output (or probability of an event occurring) and the inputs (or features)?

Graphically (for each input variable):

Logistic Regression Example: *Prediction and Understanding*

We want to predict whether someone will buy a can of tuna fish. What does the following model tell you?

$$\ln\left(\frac{prob}{1-prob}\right) = 1.2 - 5.1 \times Price + 1.8 \times Feature$$

Using Excel answer the following questions:

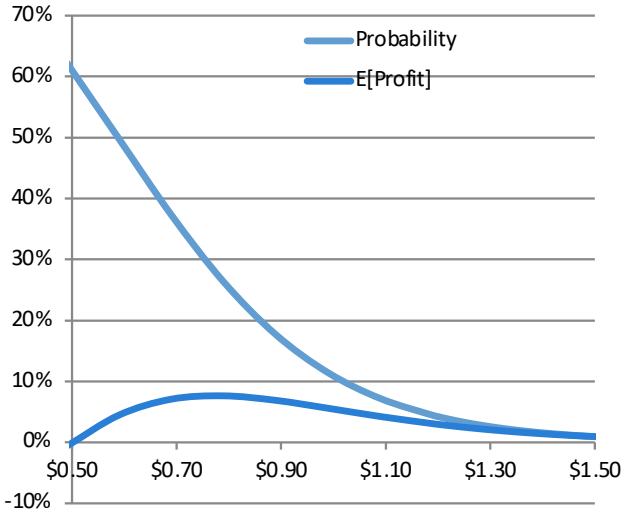
1. How do we predict the probability of purchase?
2. What is the probability of purchase if Price=$.89 and the product is on Feature=1?
3. How does changing price by $.10 effect probability of purchase?
4. What is the optimal retail price if wholesale cost is $.50?

Logistic Regression Example: *Solution*

We want to predict whether someone will buy a can of tuna fish. What does the following model tell you?

$$\ln\left(\frac{prob}{1-prob}\right) = 1.2 - 5.1 \times Price + 1.8 \times Feature$$

Price	Score	Probability	OddsRatio	E[Profit]
$ 0.49	0.501	62%	1.65	$ (0.006)
$ 0.59	-0.009	50%	0.99	$ 0.045
$ 0.69	-0.519	37%	0.60	$ 0.071
$ 0.79	-1.029	26%	0.36	$ 0.076
$ 0.89	-1.539	18%	0.21	$ 0.069
$ 0.99	-2.049	11%	0.13	$ 0.056
$ 1.09	-2.559	7%	0.08	$ 0.042
$ 1.19	-3.069	4%	0.05	$ 0.031



Logistic Regression Example: *Understanding Odds Ratios*

What is an odds ratio?

$$\text{Odds Ratio} \equiv OR = \frac{\Pr(y_i = 1)}{\Pr(y_i = 0)} = \frac{p}{1-p} = \frac{p}{q}$$

Notice to go from odds ratio to probability we compute:

$$p = \frac{OR}{1+OR}$$

Examples:

- Odds are equal then p=50%
- Odds are 4 to 1 then p=80%
- Odds are 1 in 100 then p=1%

p	*OR*
0	0
.05	.052
.3333	0.5
.50	1
.6667	2
.95	19
.999	999

Training Models
How do we find the parameters ($\boldsymbol{\beta}$)?

Our model contains both our input variables (which we know) but also *parameters* (which we do not know).

$$\mathrm{E}\left[y \mid x \right] = \mathrm{f}(x) = \mathrm{f}(x; \boldsymbol{\beta})$$

The parameters are critical to understanding response, since they tend to scale and weight the input values. Notice the parameters are fixed, which is often why they are "dropped" from the function.

Our challenge is determining the values of the parameters. This is the process of *training* the model or estimating the parameters.

The estimates of our parameters come by solving an optimization problem that measures fit, likelihood or deviance:

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left(y_i - \hat{y}_i \right)^2 = \min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left(y_i - f(\mathbf{x}_i; \boldsymbol{\beta}) \right)^2$$

Evaluating Models
Which model is best?

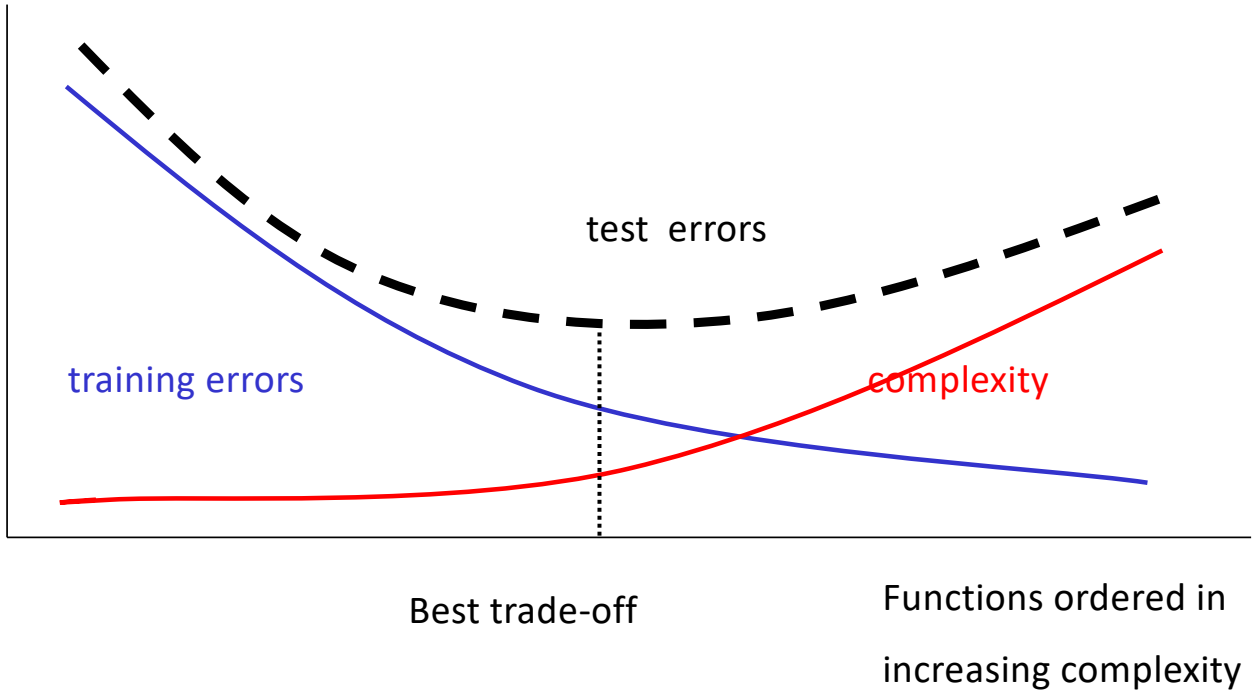
We evaluate our models based upon fit, accuracy, precision, recall, confusion matrices, lift, costs, … or some other metric that reflects our decision.

To make sure we have a model that will perform well in the future we run quasi-experiments, in which we split our dataset into a training set and test set and then only use the test set for model comparison, evaluation, and selection. This avoids overfitting.

Training Errors (Fit) versus Model Complexity (Prediction)

Min. number of training errors,

Model complexity



Communicating the Results

Interpretation and communication of the results is the last step in our data mining process. This step is crucial to a data scientist, since we are interested in having an impact on decision making.

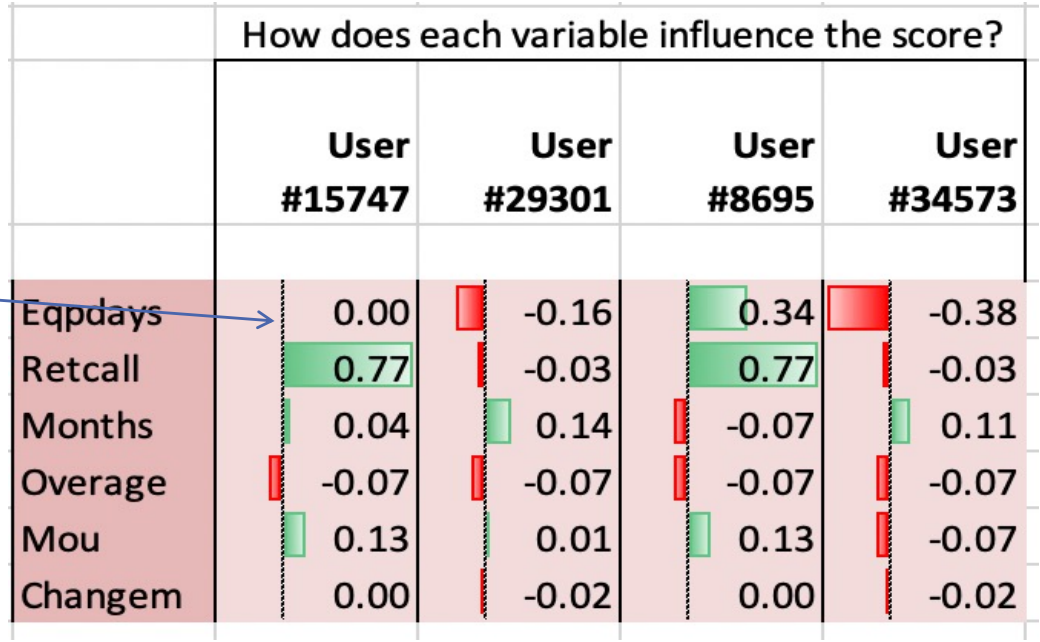
How do we do it?

Data scientists need to have the following skills

- Commitment
- Creativity
- Business savvy
- Presentation
- Intuition

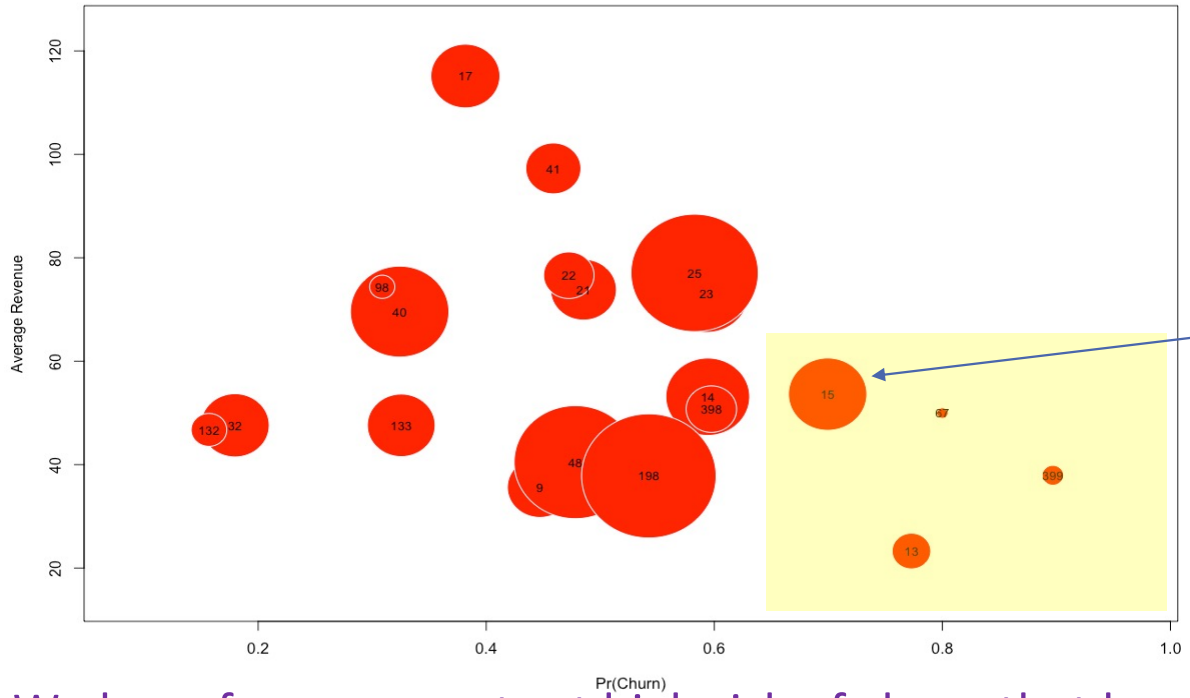
Logistic Regression Example: *Understanding Variable Influence*

For User #1
Contribution of
Eqpdays
= Beta * (Value-Mean)
= .0011 * (391-392.8)
= 0.00



Logistic regression is “compensatory”. Notice that a negative component (relative to mean) reduces probability but this can be offset by a positive component

Logistic Regression Example:
Create "stories" from your models



- New customers whose contracts are coming due in the next two months
- Live in Detroit, Miami, Minneapolis, Ohio, Phoenix, Seattle

Story: We have four segments at high risk of churn that have lower revenue. We now know that we should target customers in Detroit whose contracts are up in the next two months and get a 20% increase in profits.

Review
What are the main lessons we have learned?

Ford Ka

- Segmentation is looking for groups of customers with similar preferences. Easier for marketers to think about groups. k-Means is useful for reducing large data sets into smaller ones.

Movie Scheduling

- We can use unstructured data (text) to compare movies (products). Illustrate when to use newer, probabilistic clustering like Topic Models versus k-Means.

Lending Club

- Business decisions usually are made by looking at profits. It is not which model predicts better, but which model leads me to a better decision

Freemium

- Good fitting models do not always yield better decisions. Sometime even "good" models yield bad recommendations – either due to correlations measured by the model attributed to something not in the model or you extrapolate. You need to figure out how to interject your knowledge into the modeling and prediction process.
- Being able to use models for business decisions requires you to tell stories and convey the meaning in non-mathematical terms.

Cell2Cell

- Integrating all our lessons together. The question we want to answer may not correspond with the data. Important to choose the correct objective (LTV over 1 year or 5 years)?

Final Thoughts

Some General Comments

Data science not an automatic process

Data science should not be taken as a “black box”

Data science makes you care more about accurate data, not less

There are many ethical and legal traps to think about

It works: data mining does help you use data better!

Where to learn more about data science

Books
- James, Witten, Hastie, Tibshirani (2017) *An Introduction to Statistical Learning*
- Geron (2019) *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow*

Online courses
- DataCamp
- Coursera
- Codecademy

R packages
- CRAN repository
- Vignettes

Websites and Blogs
- KDNuggets
- DataScienceCentral
- TowardsDataScience
- DataSciencePlus
- SmartDataCollective

Contests
- Kaggle provides code & data for 19K public datasets

Quick list of useful R packages

https://support.rstudio.com/hc/en-us/articles/201057987-Quick-list-of-useful-R-packages

To load data

- DBI - The standard for for communication between R and relational database management systems. Packages that connect R to databases depend on the DBI package.
- odbc - Use any ODBC driver with the odbc package to connect R to your database. Note: RStudio professional products come with professional drivers for some of the most popular databases.
- RMySQL, RPostgresSQL, RSQLite - If you'd like to read in data from a database, these packages are a good place to start. Choose the package that fits your type of database.
- XLConnect, xlsx - These packages help you read and write Micorsoft Excel files from R. You can also just export your spreadsheets from Excel as .csv's.
- foreign - Want to read a SAS data set into R? Or an SPSS data set? Foreign provides functions that help you load data files from other programs into R.
- haven - Enables R to read and write data from SAS, SPSS, and Stata.

R can handle plain text files – no package required. Just use the functions read.csv, read.table, and read.fwf. If you have even more exotic data, consult the CRAN guide to data import and export.

To manipulate data

- dplyr - Essential shortcuts for subsetting, summarizing, rearranging, and joining together data sets. dplyr is our go to package for fast data manipulation.
- tidyr - Tools for changing the layout of your data sets. Use the gather and spread functions to convert your data into the tidy format, the layout R likes best.
- stringr - Easy to learn tools for regular expressions and character strings.
- lubridate - Tools that make working with dates and times easier.

To visualize data

- ggplot2 - R's famous package for making beautiful graphics. ggplot2 lets you use the grammar of graphics to build layered, customizable plots.
- ggvis - Interactive, web based graphics built with the grammar of graphics.
- rgl - Interactive 3D visualizations with R
- htmlwidgets - A fast way to build interactive (javascript based) visualizations with R. Packages that implement htmlwidgets include:
- leaflet (maps)
- dygraphs (time series)
- DT (tables)
- diagrammeR (diagrams)
- network3D (network graphs)
- threeJS (3D scatterplots and globes).
- googleVis - Let's you use Google Chart tools to visualize data in R. Google Chart tools used to be called Gapminder, the graphing software Hans Rosling made famous in his TED talk.

Quick list of useful R packages

https://support.rstudio.com/hc/en-us/articles/201057987-Quick-list-of-useful-R-packages

For Spatial data

- sp, maptools - Tools for loading and using spatial data including shapefiles.
- maps - Easy to use map polygons for plots.
- ggmap - Download street maps straight from Google maps and use them as a background in your ggplots.

For Time Series and Financial data

- zoo - Provides the most popular format for saving time series objects in R.
- xts - Very flexible tools for manipulating time series data sets.
- quantmod - Tools for downloading financial data, plotting common charts, and doing technical analysis.

To write high performance R code

- Rcpp - Write R functions that call C++ code for lightning fast speed.
- data.table - An alternative way to organize data sets for very, very fast operations. Useful for big data.
- parallel - Use parallel processing in R to speed up your code or to crunch large data sets.

To work with the web

- XML - Read and create XML documents with R
- jsonlite - Read and create JSON data tables with R
- httr - A set of useful tools for working with http connections

To write your own R packages

- devtools - An essential suite of tools for turning your code into an R package.
- testthat - testthat provides an easy way to write unit tests for your code projects.
- roxygen2 - A quick way to document your R packages. roxygen2 turns inline code comments into documentation pages and builds a package namespace.
- You can also read about the entire package development process online in Hadley Wickham's R Packages book

Curated list of Awesome R packages and tools
https://github.com/qinwf/awesome-R

Awesome R
- 2019
- 2018
- 2017
- Integrated Development Environments
- Syntax
- Data Manipulation
- Graphic Displays
- HTML Widgets
- Reproducible Research
- Web Technologies and Services
- Parallel Computing
- High Performance
- Language API
- Database Management
- Machine Learning
- Natural Language Processing
- Bayesian
- Optimization
- Finance
- Bioinformatics and Biostatistics
- Network Analysis
- Spatial
- R Development
- Logging
- Data Packages
- Other Tools
- Other Interpreters
- Learning R

Resources
- Websites
- Books
- Podcasts
- Reference Cards
- MOOCs
- Lists
- R Ecosystems

Other Awesome Lists

GRAPHIC DISPLAYS

Packages for showing data.

- ggplot2 💛 - An implementation of the Grammar of Graphics.
- ggfortify - A unified interface to ggplot2 popular statistical packages using one line of code.
- ggrepel - Repel overlapping text labels away from each other.
- ggalt - Extra Coordinate Systems, Geoms and Statistical Transformations for ggplot2.
- ggstatsplot - ggplot2 Based Plots with Statistical Details
- ggtree - Visualization and annotation of phylogenetic tree.
- ggtech - ggplot2 tech themes and scales
- ggplot2 Extensions - Showcases of ggplot2 extensions.
- lattice - A powerful and elegant high-level data visualization system.
- corrplot - A graphical display of a correlation matrix or general matrix. It also contains some algorithms to do matrix reordering.
- rgl - 3D visualization device system for R.
- Cairo - R graphics device using cairo graphics library for creating high-quality display output.
- extrafont - Tools for using fonts in R graphics.
- showtext - Enable R graphics device to show text using system fonts.
- animation - A simple way to produce animated graphics in R, using ImageMagick.
- gganimate - Create easy animations with ggplot2.
- misc3d - Powerful functions to deal with 3d plots, isosurfaces, etc.
- xkcd - Use xkcd style in graphs.
- imager - An image processing package based on CImg library to work with images and display them.
- hrbrthemes - Opinionated, typographic-centric ggplot2 themes and theme components.
- waffle - Make waffle (square pie) charts in R.
- dendextend - visualizing, adjusting and comparing trees of hierarchical clustering.
- r2d3 - R Interface to D3 Visualizations
- Patchwork - Combine separate ggplots into the same graphic.
- plot3D - Plotting Multi-Dimensional Data
- plot3Drgl - Plotting Multi-Dimensional Data - Using 'rgl'

Conclusion

Data science is a systematic way of using data to find knowledge and make better decisions.

Data mining is about finding patterns in data. There are many techniques, some are specialized and others are general. Each requires some judgment by the analyst to prepare the data and interpret the results.

Data mining is not an end but a means to an end. Managers are interested in taking action upon these findings. We must be able to translate data mining techniques into action.

Some final advice

Don’t be afraid to use what you have learned.

You know enough to be data scientists

…so go out and do some data mining!