Technology Fundamentals for Analytics Lab

Jason Kuruzovich

Agenda

1. Review Last time
2. R vs Python
3. Data preparation in Python
4. Analysis with Python

iPython Notebook Viewer http://nbviewer.ipython.org

or from a command line: `ipython notebook` from directory containing notebooks

(To install necessary files /scripts/4_ipython.sh)

Getting Started with Python

Introduction to Unix
Introduction to Python
Introduction to Python 2
Regular Expressions

Unix Scripting can be a useful way of splitting, searching and preprocessing text files.

Unix Scripting: Creating a new file

We will talk about sending the output of one command to another below (“pipes”), but an important command-line operator is the “redirection” operator “>”. With “>” you can send the result of your command-line processing to a file. So if you’re using grep (described next) to find all the lines that contain “foo”, you can create a new file with just these lines using redirection:

```
grep 'foo' orig_file.txt > new_file.txt
```

Unix Scripting: Grep

A utility for pattern matching. grep is by far the most useful unix utility. While grep is conceptually very simple, an effective developer or data scientist will no doubt find themselves using grep dozens of times a day. grep is typically called like this:

```
grep [options] [pattern] [files]
grep 'foo bar' sample.txt #Match all in file
grep -v 'foo bar' sample.txt #Inverse Matching
grep -R 'hee haw' . #Recursive matching. Here grep descends sub folders.
```

R vs Python - Some thoughts from around the web

- "The main advantage of Python over R is that it's a real programming language in the C family. It scales easily, so it's conceivable that anything you have in your sandbox can be used in production."
- "I use both Python (for data analysis ofcourse including numpy and scipy) and R next to each other. However, I use R exclusively to perform data analysis, and Python for more generic programming tasks (e.g. workflow control of a computer model)."

R vs Python - Some thoughts from around the web

- "Many of the commenters brought up the fact that R, while maybe not as fast (although that too is debatable) is much better for data analysis because of the huge number of libraries, tests, and its syntactical advantages (i.e. using formulas)."

iPython Notebooks

- The IPython Notebook is a web-based interactive computational environment where you can combine code execution, text, mathematics, plots and rich media into a single document
- Use `ipython notebook` to launch
- (In the /scripts folder there is a script to install).

*iPython - (1) Basics *

- The IPython Notebook

*iPython - (2) More on Python *

- The IPython Notebook

*iPython - (3) Regular Expressions *

- The IPython Notebook

Assignment

Public solutions/tutorials to Kaggle problems can be tremendous opportunities to learn data science. Try goggling "Kaggle tutorial" or "Kaggle Solutions" through Google and github.

Solution Assessment and Analysis.

The goal of the first assignment is to understand solutions to analytics problems. Overall, you should assess 3 total solutions:

Solutions 1-2 should be for the Titanic, and you should be able to get a prediction for each. You should compare differences in the solutions (in terms of performance) so both solutions must work. You can choose from Python or R based solutions (and other languages with approval).

1. Overview of Titanic Analytics Problem and Data (1 page)
2. Solutions. Attempt to detail 2 detailed solutions Include the source of the author, the analytical approach. Provide an overview and include outcome in appendix.
3. Compare the predictive performance.

Solution 3: Open to whichever solution you choose.

1. Overview of Analytics Problem and Data (1 page)
2. Provide overview of solution/approach.
3. Prepare 3 minute presentation on solution.

Take some time and find at least 2 potential solutions to Kaggle problems [Post to Canvas->Pages->Google Doc]

Titanic: A Sample Case

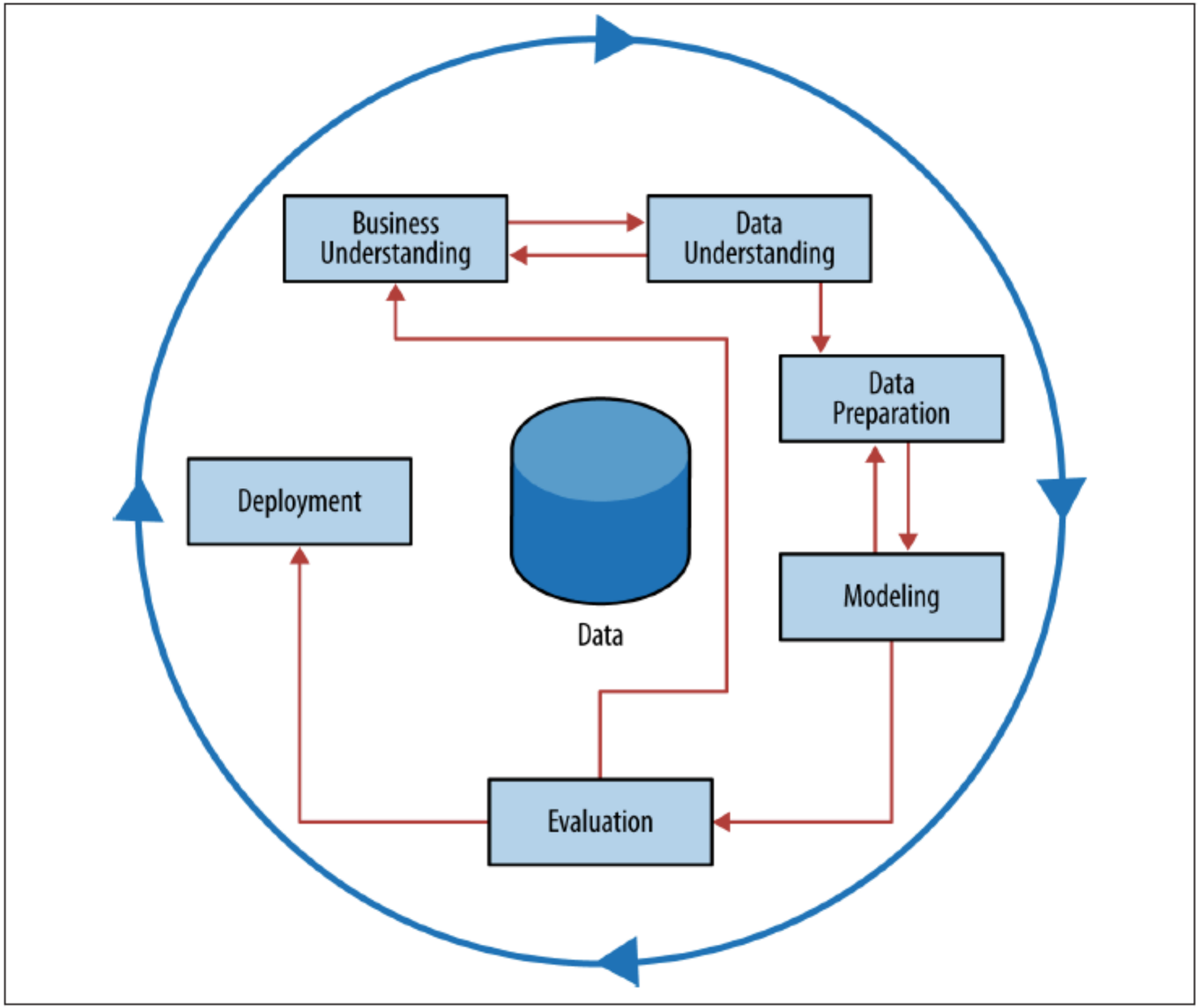

Figure 2-2. The CRISP data mining process.

Cross Industry Standard Process for Data Mining (CRISP-DM; Shearer, 2000),

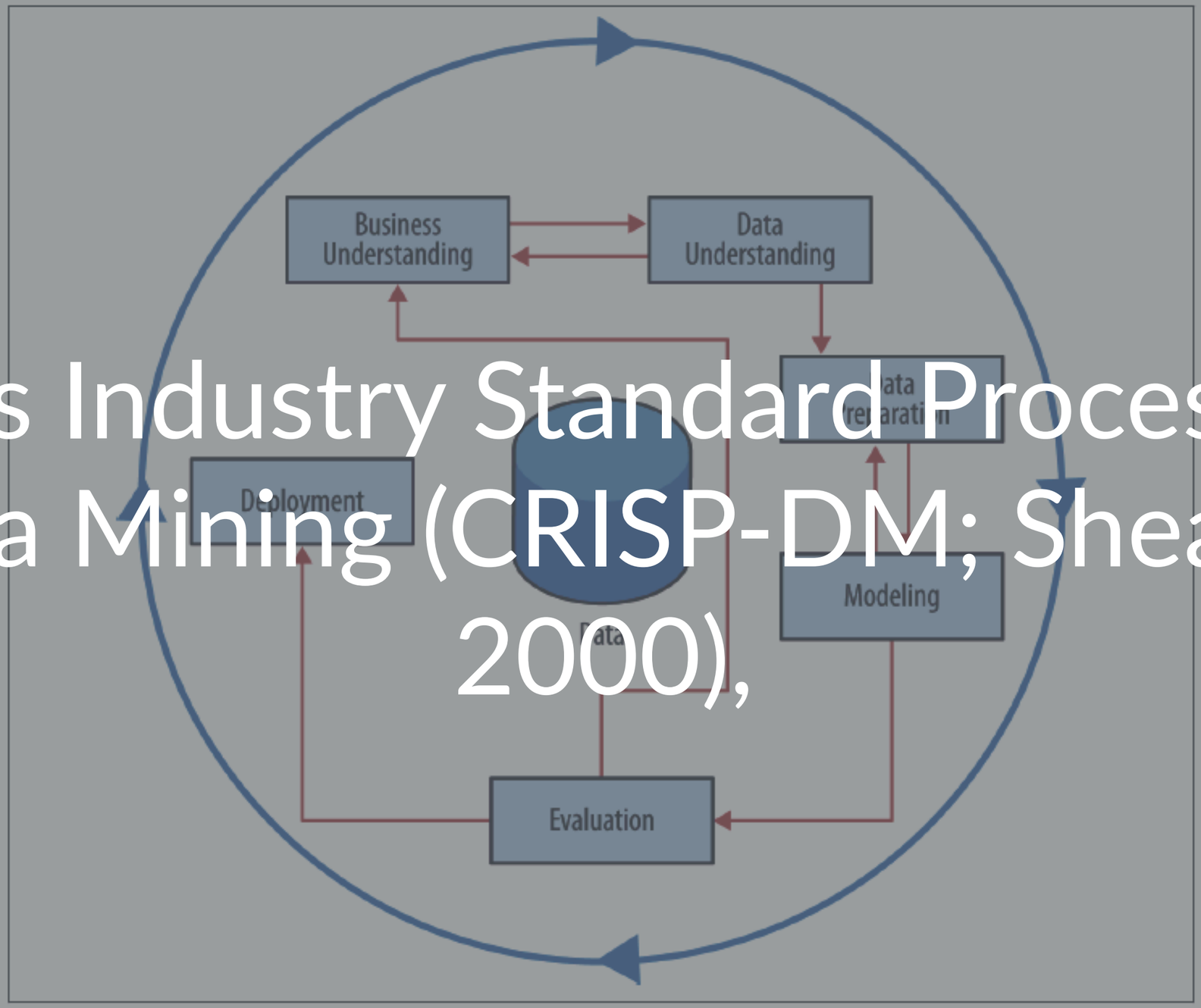

Figure 2-2. The CRISP data mining process.

Stages of Model Development

Pay attention we will use this as a framework

1. Data understanding
2. Data preparation
3. Modeling
4. Evaluation
5. Deployment (DDD)
5. Business Understanding

Titanic: Case Description

The sinking of the RMS Titanic is one of the most infamous shipwrecks in history. On April 15, 1912, during her maiden voyage, the Titanic sank after colliding with an iceberg, killing 1502 out of 2224 passengers and crew.

One of the reasons that the shipwreck led to such loss of life was that there were not enough lifeboats for the passengers and crew. Although there was some element of luck involved in surviving the sinking, some groups of people were more likely to survive than others, such as women, children, and the upper-class.

Kaggle Case (https://www.kaggle.com/c/titanic-gettingStarted)

Data Files

File Name	Available Formats
train	.csv (59.76 kb)
gendermodel	.csv (3.18 kb)
genderclassmodel	.csv (3.13 kb)
test	.csv (27.96 kb)
gendermodel	.py (3.58 kb)
genderclassmodel	.py (5.63 kb)
myfirstforest	.py (3.99 kb)

Titanic: Data Understanding

Data Files

File Name	Available Formats
train	.csv (59.76 kb)
gendermodel	.csv (3.18 kb)
genderclassmodel	.csv (3.18 kb)
test	.csv (27.96 kb)
gendermodel	.py (3.58 kb)
genderclassmodel	.py (5.63 kb)
myfirstforest	.py (3.99 kb)

What is the difference between the train and the test data?

What variable are we trying to predict?

What should be provided to Kaggle?

What should be provided to Kaggle?

PassengerId,Survived
892,0
893,1
894,0
895,0
896,1
897,0
898,1
899,0
900,1

Titanic: Data Understanding

VARIABLE DESCRIPTIONS:
PassengerId Unique Identifier
survival Survival (0 = No; 1 = Yes)
pclass Passenger Class (1 = 1st; 2 = 2nd; 3 = 3rd)
name Name
sex Sex
age Age

Titanic: Data Understanding (continued)

VARIABLE DESCRIPTIONS:
sibsp Number of Siblings/Spouses Aboard
parch Number of Parents/Children Aboard
ticket Ticket Number
fare Passenger Fare
cabin Cabin
embarked Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)

Titanic: Data Understanding (continued)

```
titanic=read.csv(file="./data/titanic_train.csv",header=TRUE,sep=",")

str(titanic)
```

```
> str(titanic)
'data.frame':   891 obs. of  12 variables:
 $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
 $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
 $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
 $ Name       : Factor w/ 891 levels "Abbing, Mr. Anthony",..: 109 191 358 277 16 559 520 629 417 581 ...
 $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
 $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
 $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
 $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
 $ Ticket     : Factor w/ 681 levels "110152","110413",..: 524 597 670 50 473 276 86 396 345 133 ...
 $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
 $ Cabin      : Factor w/ 148 levels "","A10","A14",..: 1 83 1 57 1 1 131 1 1 1 ...
 $ Embarked   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
> 
```

Titanic: Data Understanding (continued)

```
titanic=read.csv(file="./data/titanic_train.csv",header=TRUE,sep=",")

str(titanic)

summary(titanic)
```

```
> summary (titanic)
  PassengerId       Survived          Pclass                                                 Name          Sex          Age       
 Min.   :  1.0   Min.   :0.0000   Min.   :1.000   Abbing, Mr. Anthony                          :  1   female:314   Min.   : 0.42  
 1st Qu.:223.5   1st Qu.:0.0000   1st Qu.:2.000   Abbott, Mr. Rossmore Edward                  :  1   male  :577   1st Qu.:20.12  
 Median :446.0   Median :0.0000   Median :3.000   Abbott, Mrs. Stanton (Rosa Hunt)             :  1                Median :28.00  
 Mean   :446.0   Mean   :0.3838   Mean   :2.309   Abelson, Mr. Samuel                          :  1                Mean   :29.70  
 3rd Qu.:668.5   3rd Qu.:1.0000   3rd Qu.:3.000   Abelson, Mrs. Samuel (Hannah Wizosky)        :  1                3rd Qu.:38.00  
 Max.   :891.0   Max.   :1.0000   Max.   :3.000   Adahl, Mr. Mauritz Nils Martin               :  1                Max.   :80.00  
                                                  (Other)                                      :885                NA's   :177    
     SibSp           Parch           Ticket         Fare               Cabin     Embarked
 Min.   :0.000   Min.   :0.0000   1601    :  7   Min.   :  0.00             :687    :  2  
 1st Qu.:0.000   1st Qu.:0.0000   347082  :  7   1st Qu.:  7.91   B96 B98   :  4   C:168  
 Median :0.000   Median :0.0000   CA. 2343:  7   Median : 14.45   C23 C25 C27:  4   Q: 77  
 Mean   :0.523   Mean   :0.3816   3101295 :  6   Mean   : 32.20   G6        :  4   S:644  
 3rd Qu.:1.000   3rd Qu.:0.0000   347088  :  6   3rd Qu.: 31.00   C22 C26   :  3          
 Max.   :8.000   Max.   :6.0000   CA 2144 :  6   Max.   :512.33   D         :  3          
                                  (Other) :852                    (Other)   :186          
```

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171	7.25		S
2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	female	38	1	0	PC 17599	71.2833	C85	C
3	1	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2. 31	7.925		S
4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	0	113803	53.1	C123	S
5	0	3	Allen, Mr. William Henry	male	35	0	0	373450	8.05		S
6	0	3	Moran, Mr. James	male		0	0	330877	8.4583		Q
7	0	1	McCarthy, Mr. Timothy J	male	54	0	0	17463	51.8625	E46	S
8	0	3	Palsson, Master. Gosta Leonard	male	2	3	1	349909	21.075		S
9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27	0	2	347742	11.1333		S
10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14	1	0	237736	30.0708		C
11	1	3	Sandstrom, Miss. Marguerite Rut	female	4	1	1	PP 9549	16.7	G6	S
12	1	1	Bonnell, Miss. Elizabeth	female	58	0	0	113783	26.55	C103	S
13	0	3	Saundercock, Mr. William Henry	male	20	0	0	A/5. 2151	8.05		S
14	0	3	Andersson, Mr. Anders Johan	male	39	1	5	347082	31.275		S
15	0	3	Vestrom, Miss. Hulda Amanda Adolfina	female	14	0	0	350406	7.8542		S
16	1	2	Hewlett, Mrs. (Mary D Kingcome)	female	55	0	0	248706	16		S
17	0	3	Rice, Master. Eugene	male	2	4	1	382652	29.125		Q
18	1	2	Williams, Mr. Charles Eugene	male		0	0	244373	13		S
19	0	3	Vander Planke, Mrs. Julius (Emelia Maria Vandemoort	female	31	1	0	345763	18		S
20	1	3	Masselmani, Mrs. Fatima	female		0	0	2649	7.225		C
21	0	2	Fynney, Mr. Joseph J	male	35	0	0	239865	26		S
22	1	2	Beesley, Mr. Lawrence	male	34	0	0	248698	13	D56	S
23	1	3	McGowan, Miss. Anna "Annie"	female	15	0	0	330923	8.0292		Q
24	1	1	Sloper, Mr. William Thompson	male	28	0	0	113788	35.5	A6	S
25	0	3	Palsson, Miss. Torborg Danira	female	8	3	1	349909	21.075		S
26	1	3	Asplund, Mrs. Carl Oscar (Selma Augusta Emilia Johan	female	38	1	5	347077	31.3875		S
27	0	3	Emir, Mr. Farred Chehab	male		0	0	2631	7.225		C
28	0	1	Fortune, Mr. Charles Alexander	male	19	3	2	19950	263	C23 C25 C27	S
29	1	3	O'Dwyer, Miss. Ellen "Nellie"	female		0	0	330959	7.8792		Q
30	0	3	Todoroff, Mr. Lalio	male		0	0	349216	7.8958		S
31	0	1	Uruchurtu, Don. Manuel E	male	40	0	0	PC 17601	27.7208		C
32	1	1	Spencer, Mrs. William Augustus (Marie Eugenie)	female		1	0	PC 17569	146.5208	B78	C
33	1	3	Glynn, Miss. Mary Agatha	female		0	0	335677	7.75		Q
34	0	2	Wheadon, Mr. Edward H	male	66	0	0	C.A. 24579	10.5		S
35	0	1	Meyer, Mr. Edgar Joseph	male	28	1	0	PC 17604	82.1708		C
36	0	1	Holverson, Mr. Alexander Oskar	male	42	1	0	113789	52		S
37	1	3	Mamee, Mr. Hanna	male		0	0	2677	7.2292		C
38	0	3	Cann, Mr. Ernest Charles	male	21	0	0	A./5. 2152	8.05		S
39	0	3	Vander Planke, Miss. Augusta Maria	female	18	2	0	345764	18		S
40	1	3	Nicola-Yarred, Miss. Jamila	female	14	1	0	2651	11.2417		C
41	0	3	Ahlin, Mrs. Johan (Johanna Persdotter Larsson)	female	40	1	0	7546	9.475		S
42	0	2	Turpin, Mrs. William John Robert (Dorothy Ann Wonr	female	27	1	0	11668	21		S
43	0	3	Kraeff, Mr. Theodor	male		0	0	349253	7.8958		C
44	1	2	Laroche, Miss. Simonne Marie Anne Andree	female	3	1	2	SC/Paris 212	41.5792		C
45	1	3	Devaney, Miss. Margaret Delia	female	19	0	0	330958	7.8792		Q
46	0	3	Rogers, Mr. William John	male		0	0	S.C./A.4. 235	8.05		S
47	0	3	Lennon, Mr. Denis	male		1	0	370371	15.5		Q

Titanic: Data Preparation

1. Deal with missing data.

   Missing data can limit the ability to generate prediction. While when doing *scientific analysis* you must be very cautious with data imputation, in prediction it is necessary.

2. Recode data to create features.

   There are lots of ways that individual variables can be recoded to be used in different ways. For example, is it better to include age as a number, or as a category (~18 may be very significant for boys with the "women and children first")

Titanic: Data Preparation (1. Missing Data)

There are a variety of models to impute data. Two simple ones are to replace with the median and to calculate the predicted value for the missing variable.

```
#replace age with with the median
titanic.train$age[is.na(titanic.train$age)] <- median(titanic.train$age, na.rm=TRUE)


#predict age based on fare,gender, siblings using regression analysis
m.age <- lm(Age ~ Fare + Sex + SibSp, data = titanic.train)
titanic.train$Age[is.na(titanic.train$Age)] <- predict(m.age, newdata = titanic.train)[is.na(titanic.train$Age)]

#Use only whether individual is a child or not
titanic.train$Child <- 0
titanic.train$Child[train$Age < 18] <- 1
```

Titanic: Data Preparation

Recode data to create features.

```
train$title <- NA
train[grep('Mr[. ]', train$name), 12]       <- 'Mr'
train[grep('Don[. ]', train$name), 12]      <- 'Don'
train[grep('Dr[. ]', train$name), 12]       <- 'Dr'
train[grep('Major[. ]', train$name), 12]    <- 'Major'
train[grep('Jonkheer[. ]', train$name), 12] <- 'Jonkheer'
train[grep('Master[. ]', train$name), 12]   <- 'Master'
train[grep('Col[. ]', train$name), 12]      <- 'Col'
train[grep('Capt[. ]', train$name), 12]     <- 'Capt'
train[grep('Mrs[. ]', train$name), 12]      <- 'Mrs'
train[grep('Mme[. ]', train$name), 12]      <- 'Mme'
train[grep('Countess[. ]', train$name), 12] <- 'Countess'
train[grep('Ms[. ]', train$name), 12]       <- 'Ms'
train[grep('Miss[. ]', train$name), 12]     <- 'Miss'
train[grep('Mlle[. ]', train$name), 12]     <- 'Mlle'
train[grep('Rev[. ]', train$name), 12]      <- 'Rev'
```

Titanic: Data Modeling and Evaluation

What do you know about what is likely to drive survival in a shipwreck?

Titanic: Data Modeling and Evaluation

1. Select data for cross valuation
2. Determine the category of data model
3. Select and run the model
4. Evaluate the performance

Titanic: Data Modeling and Evaluation (Cross Validation)

Modeling optimizes the parameters to fit the training data as well as possible. If we then take an independent sample of validation data from the same population as the training data, it will generally turn out that the model does not fit the validation data as well as it fits the training data. This is called overfitting, and is particularly likely to happen when the size of the training data set is small, or when the number of parameters in the model is large. Cross-validation is a way to predict the fit of a model to a hypothetical validation set when an explicit validation set is not available.

Wikipedia

Titanic: Data Modeling and Evaluation (Cross Validation)

1. Holdout Sample (2-fold cross validation) (large datasets)
2. K-fold cross validation (large/medium datasets)
3. Leave P out cross validation (small datasets)

Titanic: Data Modeling and Evaluation (Cross Validation - Holdout Sample/2-Fold)

For each fold, we randomly assign data points to two sets d0 and d1, so that both sets are equal size (this is usually implemented by shuffling the data array and then splitting it in two). We then train on d0 and test on d1, followed by training on d1 and testing on d0.

Titanic: Data Modeling and Evaluation (Cross Validation - K-fold cross validation)

Original sample is randomly partitioned into k equal size subsamples. Of the k subsamples, a single subsample is retained as the validation data for testing the model, and the remaining k − 1 subsamples are used as training data. The cross-validation process is then repeated k times (the folds), with each of the k subsamples used exactly once as the validation data. The k results from the folds can then be averaged (or otherwise combined) to produce a single estimation. The advantage of this method over repeated random sub-sampling (see below) is that all observations are used for both training and validation, and each observation is used for validation exactly once. 10-fold cross-validation is commonly used,

Titanic: Data Modeling and Evaluation (Cross Validation - Leave P out cross validation)

Leave-p-out cross-validation (LpO CV) involves using p observations as the validation set and the remaining observations as the training set. This is repeated on all ways to cut the original sample on a validation set of p' observations and a training set. LpO cross-validation requires to learn and validate C_n^p times (where n is the number of observation in the original sample). So as soon as n is quite big it becomes impossible to calculate.

Titanic: Data Modeling and Evaluation

Determine the category of data model. We need a classifier. (This is just a sampling).

1. Simple prediction
2. Logistic Regression [only good for 2 categories]
3. Random Forest

Titanic: Data Modeling and Evaluation

(python) This uses a simple loop.

```
# Finally, loop through each row in the train file, and look in column index [3] (which is 'Sex')
# Write out the PassengerId, and my prediction.

predictions_file = open("gendermodel.csv", "wb")
predictions_file_object = csv.writer(predictions_file)
predictions_file_object.writerow(["PassengerId", "Survived"])   # write the column headers
for row in test_file_object:                                    # For each row in test file,
    if row[3] == 'female':                                      # is it a female, if yes then
        predictions_file_object.writerow([row[0], "1"])         # write the PassengerId, and predict 1
    else:                                                       # or else if male,
        predictions_file_object.writerow([row[0], "0"])         # write the PassengerId, and predict 0.
test_file.close()                                               # Close out the files.
predictions_file.close()
```

Titanic: Data Modeling and Evaluation

Logistic Model

```
logistic.model <- glm(survived ~ pclass + sex, family = binomial(), data=train)

#generate predictions for training data using the predict method of the logistic model
 training_predictions <- predict(logistic.model, type = "response")
 test_predictions[test_predictions >=0.5] <- 1
 test_predictions[ test_predictions != 1] <- 0
 test_predictions[is.na(test_predictions)] <- 0
```

Titanic: Data Modeling and Evaluation

Random Forest

```
train.rf <- randomForest(formula<-survived~pclass+sex+age_imp+sibsp+parch+fare+immature+noble+cabin_pos+cabin_floor+ticket_no+line, data=train)
print(train.rf)
train.rf$importance
varImpPlot(train.rf)
pred <- predict(train.rf, test)
```

For each passenger in the test set, you must predict whether or not they survived the sinking (0 for deceased, 1 for survived). Your score is the percentage of passengers you correctly predict. What is an appropriate baseline for estimating performance?

Random guess. A coin flip decides whether you pick if someone survives or not.

How could we do better without building a model?

Everyone dies. Because only 38% of the people survive, we can beat a coin flip by predicting everyone dies.

Two other good tutorials

Python Tutorial
R Tutorial