R for Scientific and Data Intensive Computing

Burak Himmetoglu
ETS & CSC
bhimmetoglu@ucsb.edu

10/11/2017

Enterprise Technology Services | University of California Santa Barbara

CNSI UCSB MRL

CSC

UNIVERSITY OF CALIFORNIA SANTA BARBARA
CENTER FOR SCIENTIFIC COMPUTING

Who uses R for what purpose?

Scientists, engineers and developers of a wide range of interests!

- Statistics
- Simulations
- Bioinformatics (Bioconductor)
- Data Analysis
- Predictive analysis, machine learning
- Data Visualization
- Web Apps, Packages, Projects (RStudio)

Question: R takes a long time to run, what can I do?

Possible answers:

- Use specialized packages for performance 😀
- Try simple (shared memory) parallel tools 😀
- Run your R code in a remote cluster 😀/😐
  - Large datasets that don’t fit your computer’s memory
  - Manually divide computations

- Try (distributed memory) parallelism, or Spark solutions 😬/😱
- Write C/C++ extensions for R 😬/😱

Examples in this seminar:

Clone the repository:

```
git clone https://github.com/bhimmetoglu/CSC-Computing-2017
```

For example on the cluster (Knot):

```
export PATH="/sw/csc/R-3.2.3/bin:$PATH"
```

Tutorial 1: Titanic Survival Prediction

https://www.kaggle.com/c/titanic

Jack:

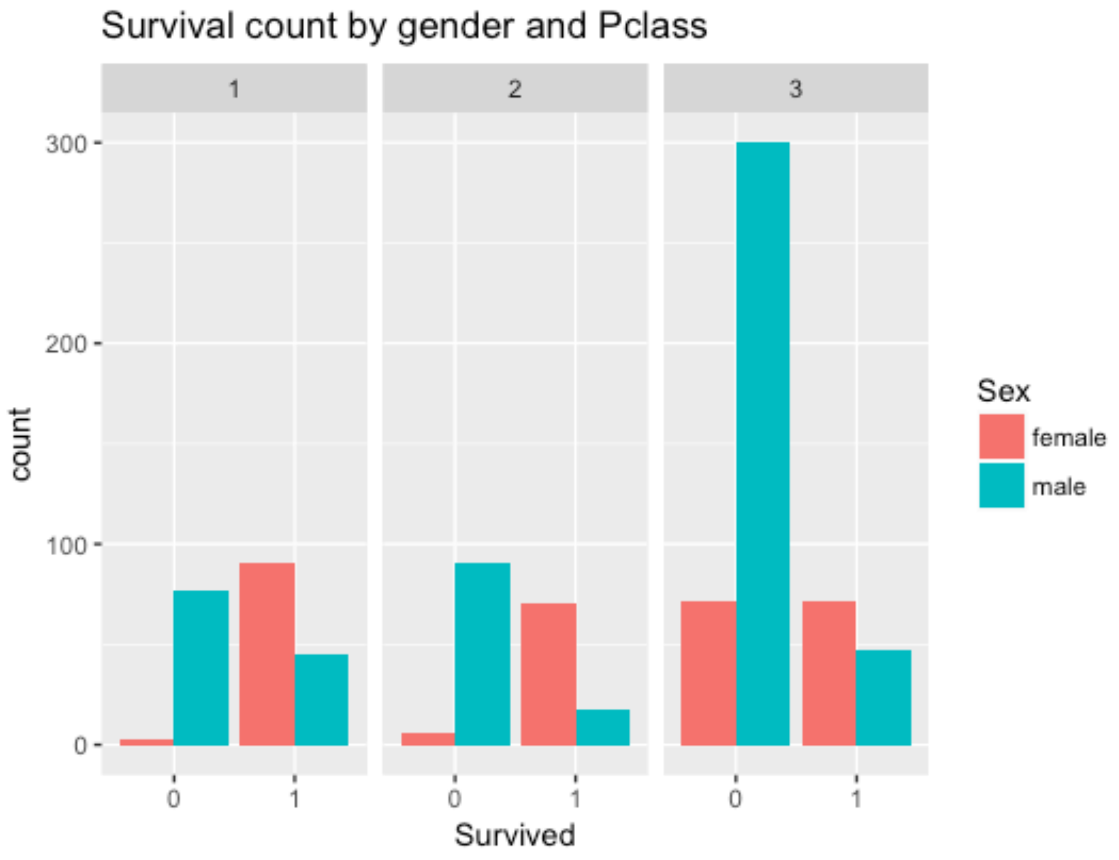
$P(Survived) \simeq 0.19$

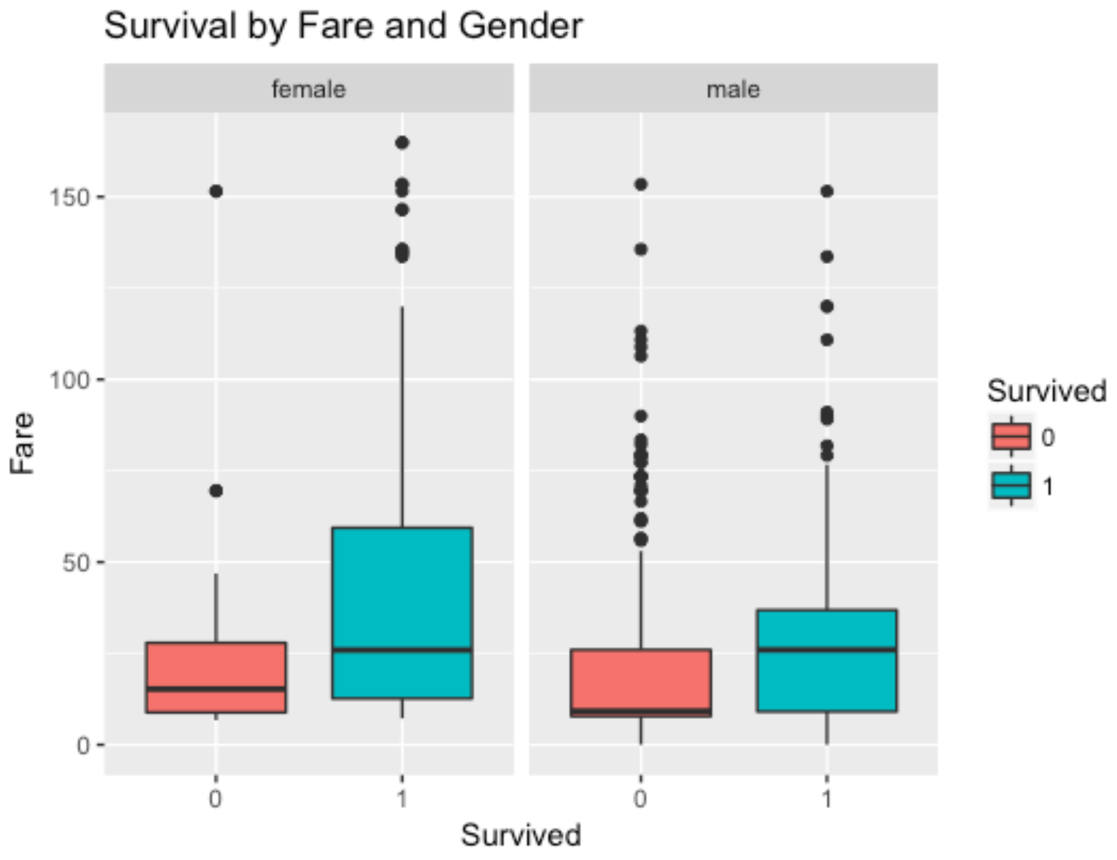
Rose:

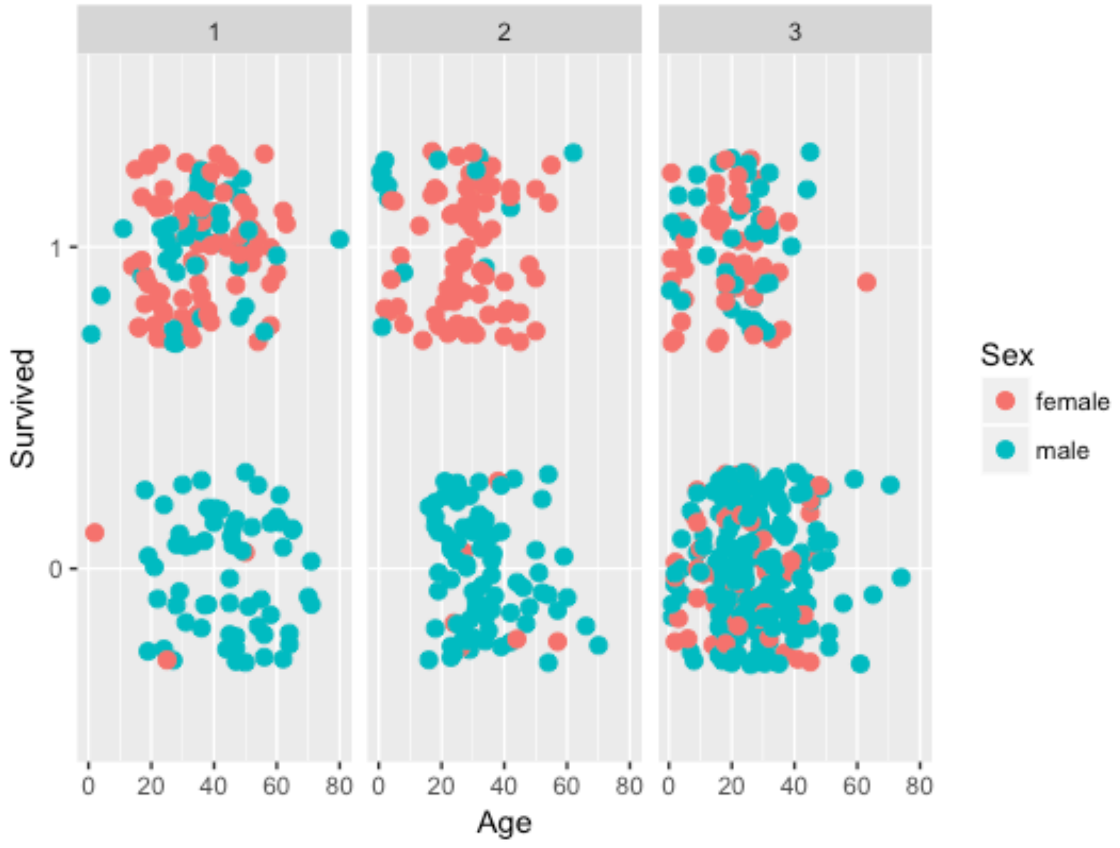
$P(Survived) \simeq 0.74$

Prediction purely based on gender

Can we predict who survived?


Survival count by gender and Pclass


Survival by Fare and Gender



Tidyverse

A collection of packages for data processing and visualization

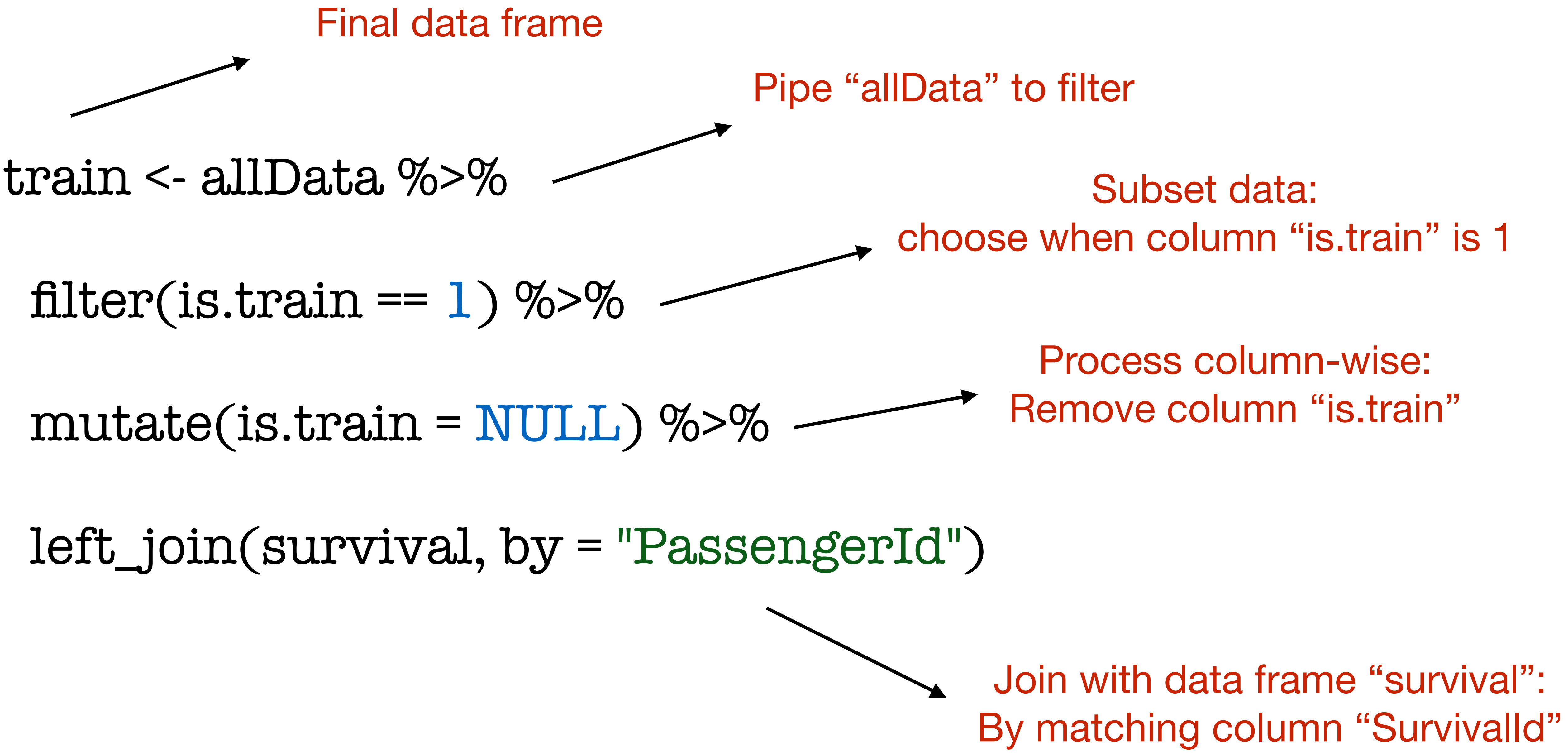
https://www.tidyverse.org

E.g.: dplyr package contains these useful functions:

• group_by( )	# group by given column
• summarize( )	# assign a new column by aggregation
• mutate( )	# create/remove/manipulate columns
• left_join( )	# join data frames
• filter( )	# filter by a given rule
• select( )	# select columns
•	

Data wrangling with: Dplyr

E.g.: Combine two data frames in a custom way. Connect operations by “pipe”



```
train <- allData %>%
  filter(is.train == 1) %>%
  mutate(is.train = NULL) %>%
  left_join(survival, by = "PassengerId")
```

Final data frame

Pipe “allData” to filter

Subset data:
choose when column “is.train” is 1

Process column-wise:
Remove column “is.train”

Join with data frame “survival”:
By matching column “SurvivalId”

Model Matrices

- We need to convert all factor variables into numeric ones
- In general, values cannot be compared
- E.g. States in U.S, Gender, City etc.

```
model.matrix()
sparse.model.matrix()
```

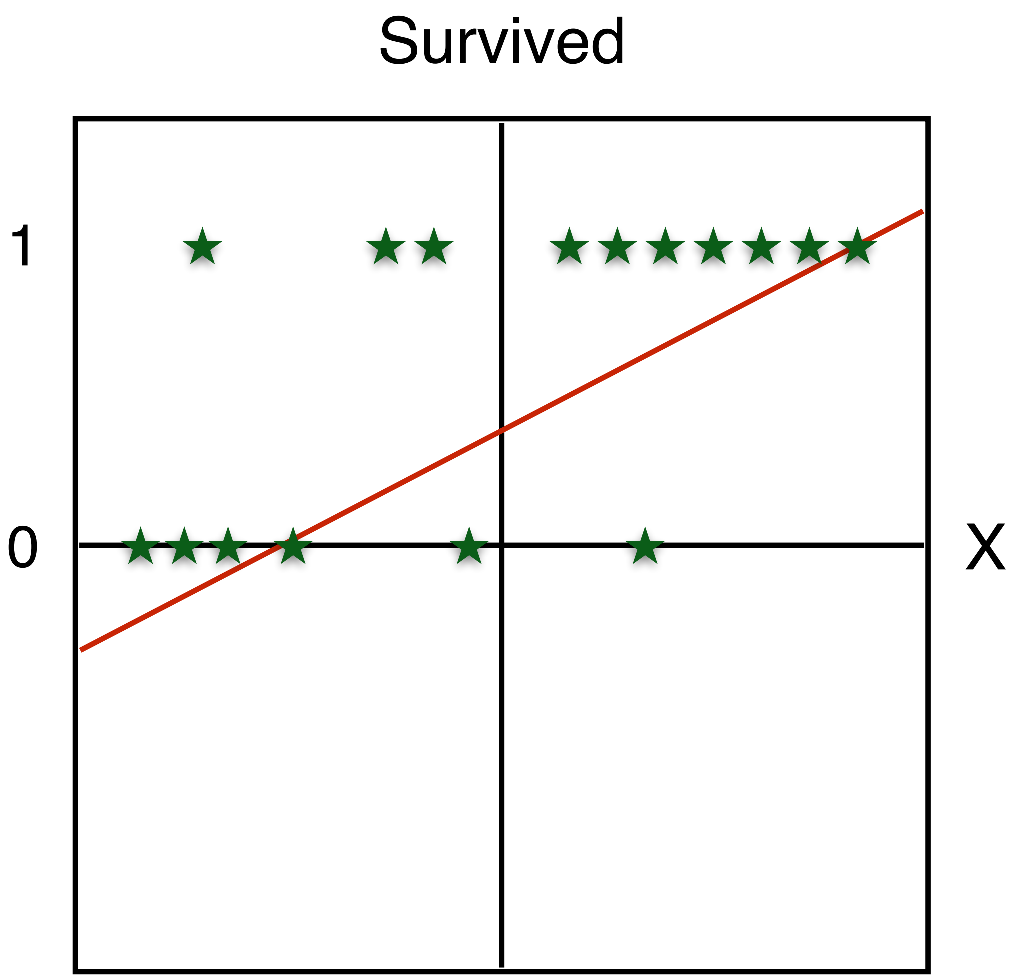
Id	Pclass	Age
1	1	45
2	2	50
3	2	22
4	3	18
5	1	65
6	2	34

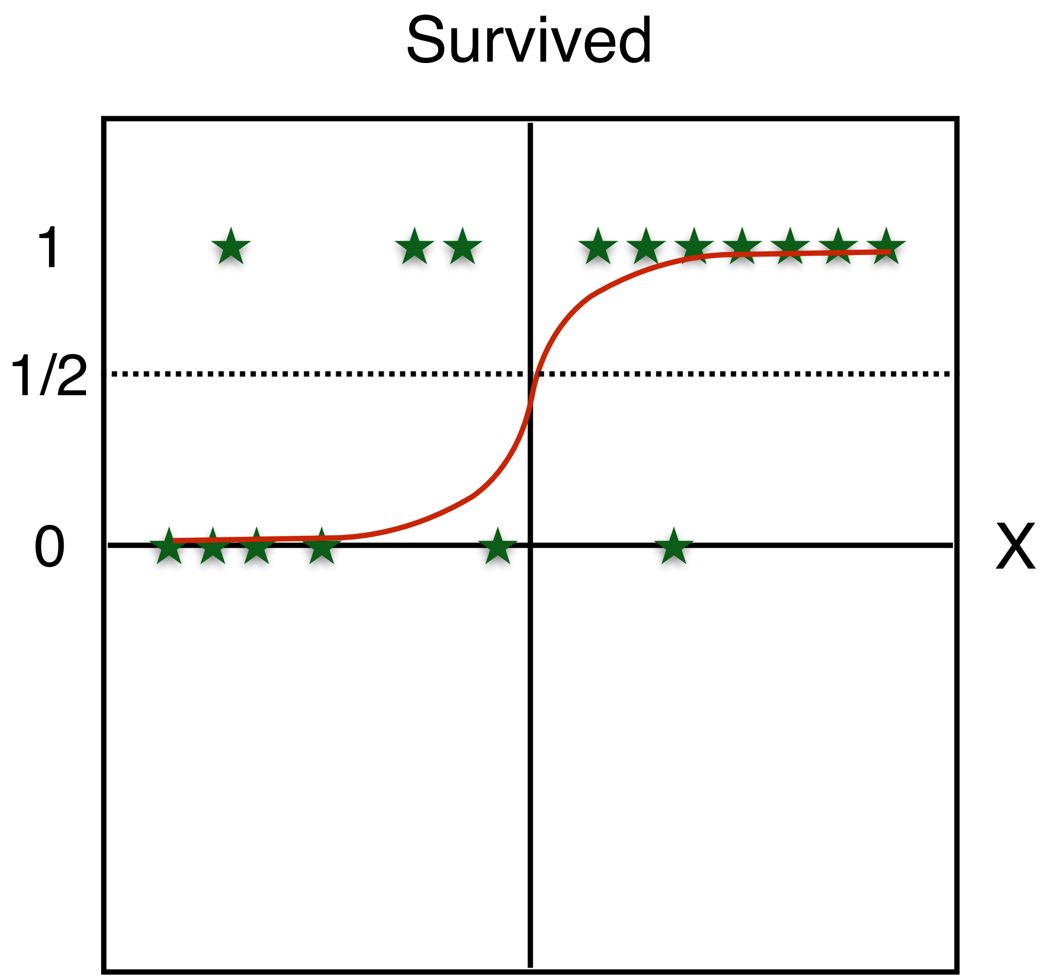
Id	Pclass2	Pcalss3	Age
1	0	0	45
2	1	0	50
3	1	0	22
4	0	1	18
5	0	0	65
6	1	0	34

Logistic Regression

- Linear model for classification

$$z_i = \beta_0 + \beta_1^T \cdot \mathbf{x}_i \qquad y_{\text{pred, i}} = \frac{1}{1 + e^{-z_i}}$$





Logistic Regression (with regularization)

- Parameters β_0, β_1 optimized to yield small error
- Overfitting problem: LASSO and Ridge regression
- α, λ by cross-validation (parallel part in glmnet)

This is the optimization problem:

$$\min_{\beta_0,\beta} \frac{1}{N} \sum_{i=1}^{N} l(y_i, \beta_0 + \beta^T x_i) + \lambda \left[(1-\alpha)||\beta||_2^2/2 + \alpha||\beta||_1 \right]$$

```
# Functions to use:
cv.glmnet() # Determines λ by cross-validation
glmnet()    # Determines β0, β1 by optimization
```

Tutorial 2: Run R code on Knot cluster

- Remember: No RStudio to experiment with!
- Make sure that your R code runs from start to end
- Perform tests on your computer first

A simple script (text file) can be used to submit to the queue:

```
#!/bin/bash
#PBS -l nodes=1:ppn=12
#PBS -l walltime=01:00:00
#PBS -N MonteCarlo
#PBS -V

cd $PBS_O_WORKDIR

Rscript --vanilla montecarlo.R > output
```

Tutorial 2: Run R code on Knot cluster

Monte Carlo integration:

$$Z = \int_0^1 \int_0^1 \dots \int_0^1 dx_1\, dx_2\, \dots dx_n\, e^{-x_1^2 - x_2^2 - \cdots - x_n^2}$$

```
For (i = 1, NumSimulations){

    Pick $\{x_1, x_2, \dots, x_n\}$ from a uniform distribution

    Z ← (Volume of region) x  Integrand at $\{x_1, x_2, \dots, x_n\}$

}
Average results (Z's)
```

Running multiple R instances concurrently

```
#!/bin/bash
#PBS -l nodes=1:ppn=12
#PBS -l walltime=01:00:00
#PBS -N MonteCarlo
#PBS -V

cd $PBS_O_WORKDIR

Rscript --vanilla part1.R > out1 &
Rscript --vanilla part2.R > out2 &
.....
Rscript --vanilla part12.R > out12 &

wait
```

Resources for learning R

- swirl package (install.packages(“swirl”))
- Coursera : https://www.coursera.org/learn/r-programming
- DataCamp: https://www.datacamp.com/courses/free-introduction-to-r

Introduction to Statistical Learning
with applications in R

http://www-bcf.usc.edu/~gareth/ISL/