DECISION PATTERNS

R Fundamentals

Practical Machine Learning (with R)

UC Berkeley

Fall 2016

Agenda

- Administrativa
  - Role Call
    - Missing Coordinates
  - November 23, 2016?
  - Class Google Group
- Review
- New Topics

Assignment – Due 10/11 11:59 PM

- **MLR** Chapter 3, Chapter 6 pp.171-200
- Introduction to dplyr .. https://cran.r-project.org/web/packages/dplyr/vignettes/introduction.html
- Introduction to data.table .. https://cran.r-project.org/web/packages/data.table/vignettes/datatable-intro.pdf
- Introducting Magrittr .. https://cran.r-project.org/web/packages/magrittr/vignettes/magrittr.html

REVIEW AND EXPECTATIONS

EXPECTATIONS: R

- You have installed **R** and **Rstudio**

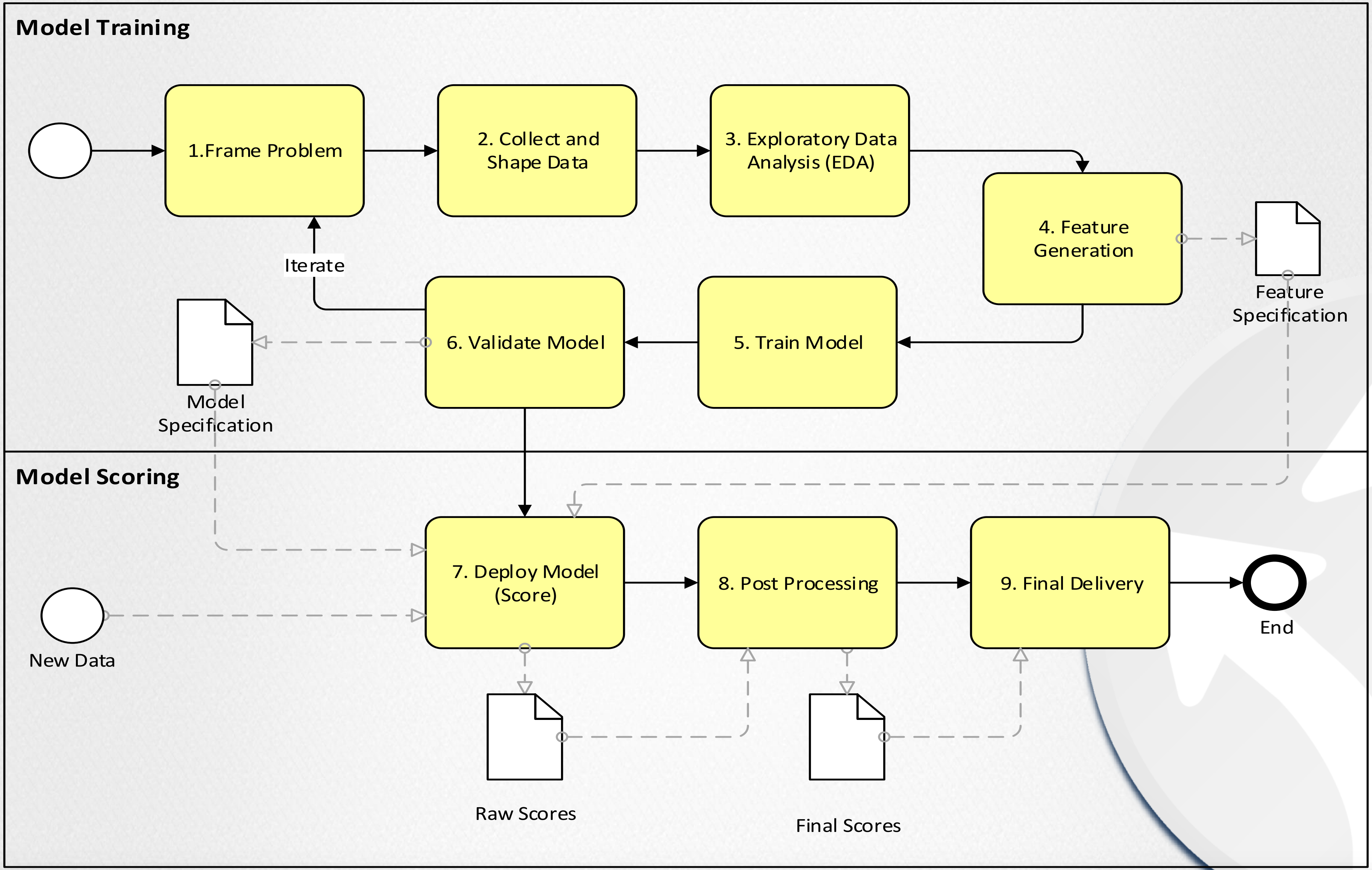
- If you are new to **R**, you will have checked out one of the resources and have started becoming familiar with syntax and functions.

EXPECTATIONS: GIT

- You understand:
  - installed **git** and created a github account
  - **fork** the class repo(sitory)
  - **clone** a local copy of the repo
  - **pull** new changes
  - edit existing files
  - **add** and **commit** changes
  - **push** the assignment back to your repo
- Now: **pull** upstream changes

`CSX460/CSX460.git`

Expectations: Process



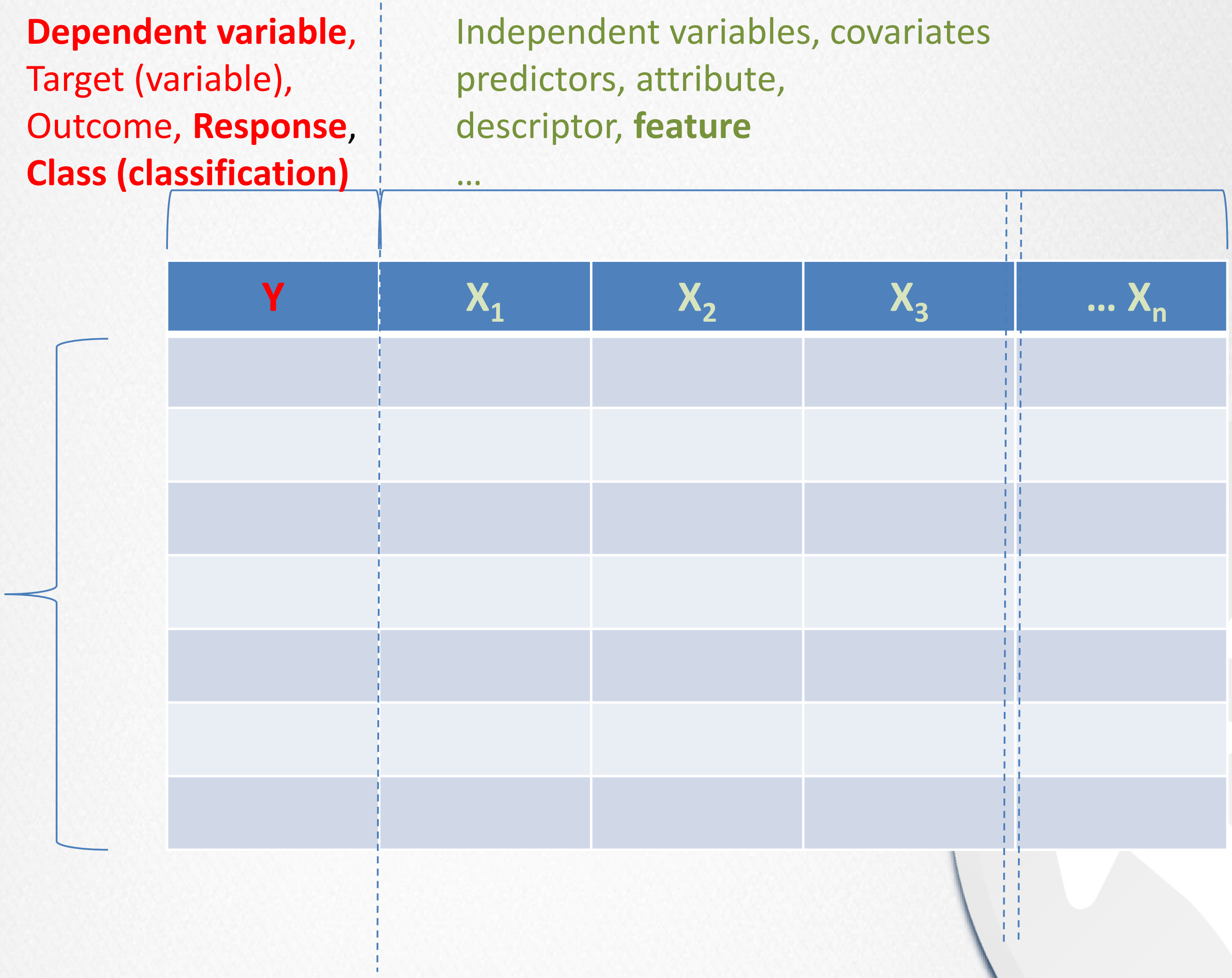
MLwR Chapter 1

- Four Parts to "Learning" Process
- Five Steps for Modeling
- **Type**s of Data
- Types of Machine Learning Algorithms

MLwR Chapter 2

- Data structures
- Saving/Loading Data With R
- Exploring the structure of the Data
  - Numeric variables
  - Categorical variables
  - Relationship Between Variables

Data Uses



Dependent variable, Target (variable), Outcome, **Response**, **Class (classification)**

Independent variables, covariates predictors, attribute, descriptor, **feature** …

Y	X_1	X_2	X_3	… X_n

Unit of observation,
Cases,
Instance,
Data Point,
Sample

$\hat{y}$

Prediction

Forecast

Estimate

…

Getting Help in Primer

Help in R	`?,help,??,apropos`
Operators	`?Arithmetic`
Control Flow	`?Control`
Rstudio Cheatsheets	`...Google`

Magrittr: Pipe operator

```
install.package(‘magrittr’)
1:10 %>% mean
1:10 %>% add(2) %>% mean

x <- 1:10
x %<>% add(2) %>% mean
```

Notes:

* Use `backpipe` package for `%<%`

Magrittr: Pipe operator

```
install.package(‘magrittr’)
1:10 %>% mean
1:10 %>% add(2) %>% mean

x <- 1:10
x %<>% add(2) %>% mean
```

Notes:

* Use `backpipe` package for `%<%`

Data.Table: Fast Data Frmes

```
install.package(‘data.table’)
data(iris)
setDT(iris)

iris[ i, j, by= , … ]
```

Note:

- see `?data.table`

DATA.TABLE: FAST DATA FRMES

```
library(‘data.table’)
data(iris)
setDT(iris)

iris[
  by=Species,
  Species != 'setosa',
  .(  sw=mean(Sepal.Width),
      sl=mean(Sepal.Length)
  )]
```

DPLYR: DATA PIPELINES

```
install.package(‘dplyr’)
data(iris)

iris  %>%
  filter( Species != "setosa")  %>%
  group_by(Species) %>%
  summarize(
    mean(Sepal.Width),
    mean(Sepal.Length)
)
```

Note:

Uses `magrittr`

DPLYR: DATA PIPELINES

```
library(‘dplyr’)
data(iris)

iris  %>%
  filter( Species != "setosa")  %>%
  group_by(Species) %>%
  summarize(
    mean(Sepal.Width),
    mean(Sepal.Length)
)
```

BACK TO MACHINE LEARNING

Machine Learning Types

- **Type** of Response:
  - Continuous → **REGRESSION**
  - Categorical* → **CLASSIFICATION**

    *Binary is a special case

- **Availability** of "labelled" Responses
  - Available → **SUPERVISED**
  - Unavailable → **UNSUPERVISED**

  - Sometimes available/inferable →**SEMI-SUPERVISED**
  - Avail. as training progresses →**ADAPTIVE/REINFORCEMENT**

GOAL FIND A FUNCTION, f

- easy to evaluate
- Takes a one or more values of inputs
- yields a single output value for each input (row)

- Output, $\hat{\boldsymbol{y}}$, should be “close to” observed values, $\mathbf{y}$:

$$\hat{\boldsymbol{y}} \sim \boldsymbol{y}$$

QUESTIONS:

- What do we mean by “close”?
- What functions are available to be used?

∞

- How do we find one? The best one?

3 Requirement for Algorithm

- A method for evaluating how well the algorithm performs (**ERRORS**)

- A restricted class of function (**MODEL**)

- A process for proceeding through the restricted class of functions to identify the functions (**SEARCH/OPTIMIZATION**)

Our Model

Naïve Model

$$\hat{y} = mean(y)$$

Our Model, a linear model:

$$\hat{y} = \beta_0 + \beta_1 x_1$$

Search / Optimization

Find the parameters minimize that minimize the loss function …

SOLVE:

$$argmin_{\beta}\, L(\boldsymbol{y}, \hat{\boldsymbol{y}})$$

$$argmin_{\beta} \sum(\boldsymbol{y} - \hat{\boldsymbol{y}})^2 \text{ (SSE)}$$

- Direct Solution (special case)
- Recursive Goal Seeking