Technische Universität München

Business Analytics

Introduction to the DMC

Decision Sciences & Systems (DSS)

Department of Informatics

TU München

Tutorial Business Analytics

Outline

Today's topics:

- Dates & Regulations for Data Mining Cup
- Grading for Data Mining Cup
- Rules of Data Mining Cup
- Evaluation
- Steps of Data Mining Cup
- Example dataset + Script
- Presentation of dataset for DMC

Tutorial Business Analytics

Dates & Regulations

Dates

- 21th of December 2017 at 1pm until 23rd of January 2018 1 pm
- Registration starts on the 21st of December 2017 at 1pm
- 8th - 12th of January 2018: DMC support tutorial
- 23rd of January 2018 1 pm: DMC Deadline
- 25th of January 2018: Announcement of the best team to present its results
- 29th and 30th of January 2018: Checkup presentations for individual groups
- 1st of February 2018: Data Mining Cup presentation in the final lecture

Regulations

- It is not allowed to cooperate outside your group
- Solutions will be checked, individual groups will be asked to present to the DMC Tutors on either 29th or 30th of January

Tutorial Business Analytics

Grading

Performance measurement is by **accuracy**

Grade will be given with respect to the achieved accuracy in the **test data** set as well as the produced code

Final Grade

If (DMC grade better than exam grade): 75% exam grade + 25% DMC grade

Else: 100% exam grade

Example:

Exam grade (2.0), DMC grade (2.7) -> Final grade: 2.0

Exam grade (2.0), DMC grade (1.0) -> Final grade: 0.75*2.0+0.25*1.0=1.75 -> 1.7

Tutorial Business Analytics

Rules of Data Mining Cup

Teams

- Team size: 1 – 3 members.
- **Teams must be built before the first submission** (teams will be fixed after first submission!).
- Each student can only be member of one team within one Data Mining Cup.

Submissions

- Maximum number of valid submissions for each DMC: 10.
- Best ranked submission, **only**, will be taken into account for the ranking.
- For reasons of traceability you must use a fixed seed of 42 (`set.seed(42)`).

Disqualification reasons:

- **Non-reproducible** submissions (submitted predictions **must be reproducible** using the submitted R script)
- **Hard-coded** classifications (even if the best ranked submission is not hard-coded!)
- **Copies** from other groups (disqualification of both teams)

Tutorial Business Analytics

Steps of Data Mining Cups

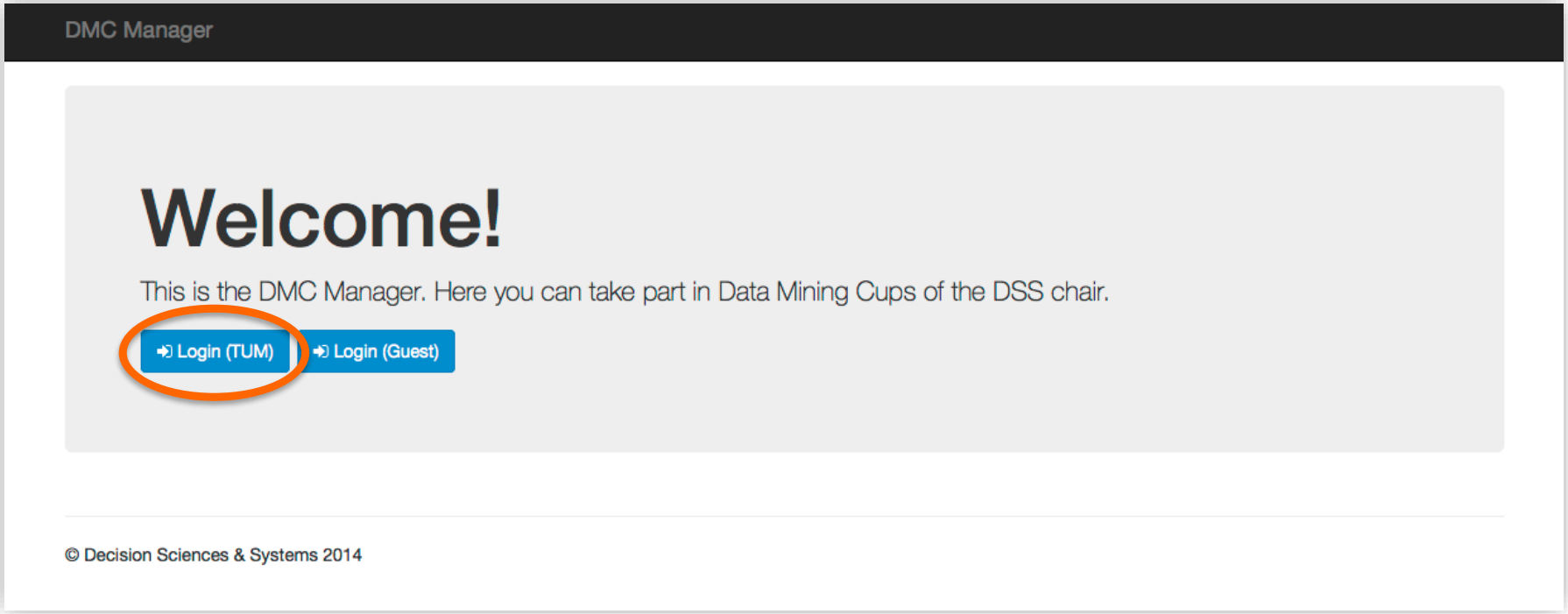
1. Build a Team in the DMC Manager
2. Load & Explore the Data Set
   - Summary statistics
   - Plotting
3. Data Preparation
   - Feature Selection
   - Discretization
4. Training & Evaluation
   - Classification Methods
   - Metrics
   - Resampling Methods
5. Predict Classes in Test Data
6. Export the Predictions
7. Upload the Predictions and the Corresponding R Script on DMC Manager

Source: http://topepo.github.io/caret/

1. Build Team in DMC Manager

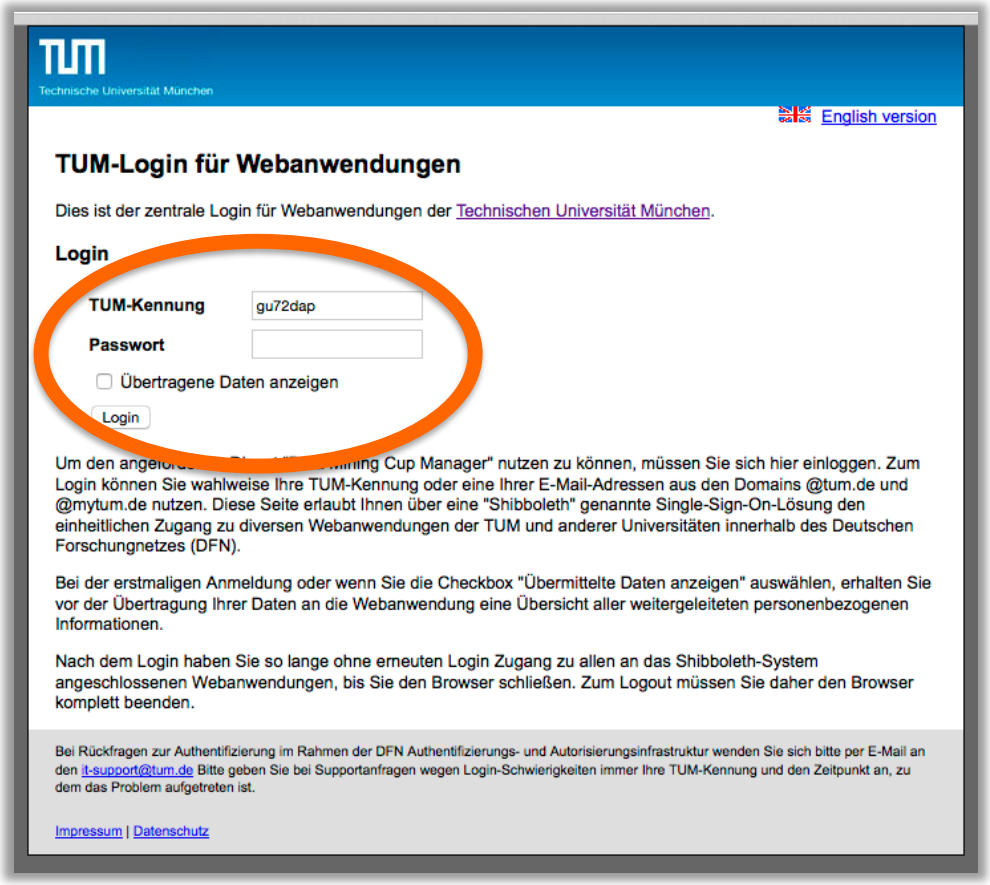
Login with your TUM login data (“TUM Kennung”)

https://dmc.dss.in.tum.de/dmc/



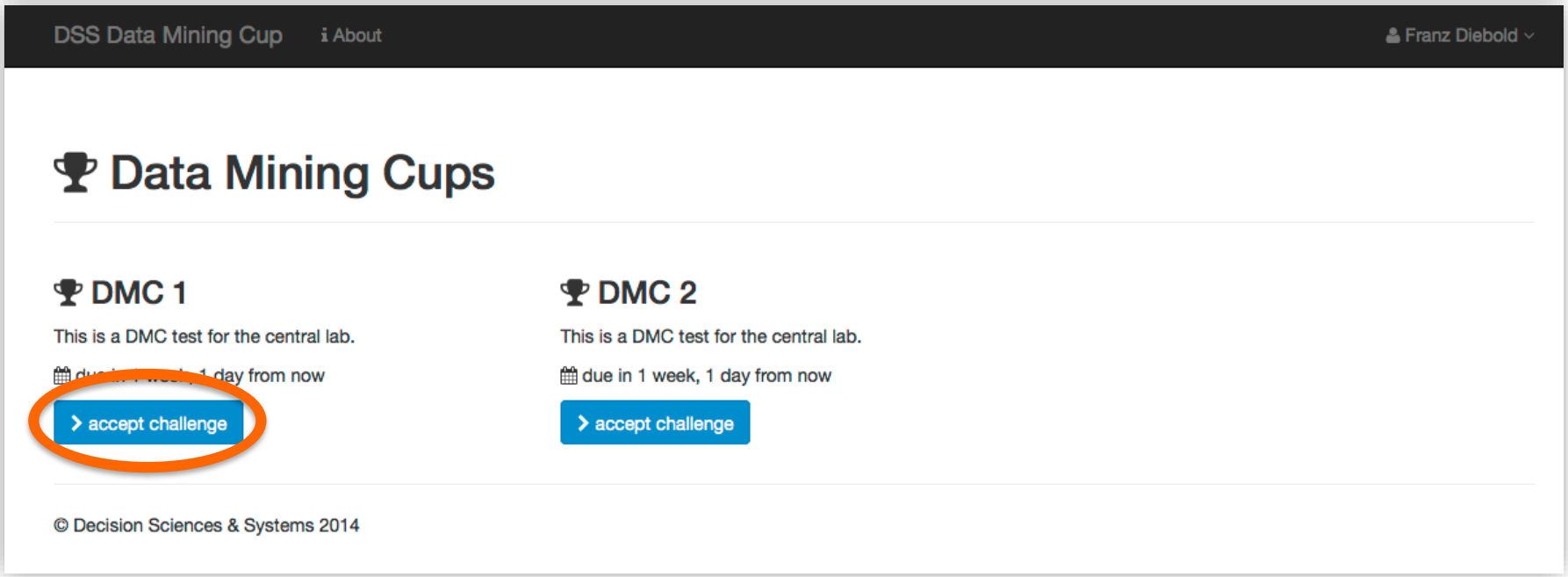
1. Build Team in DMC Manager

Login via “Shibboleth” with your TUM login data (“TUM Kennung”)



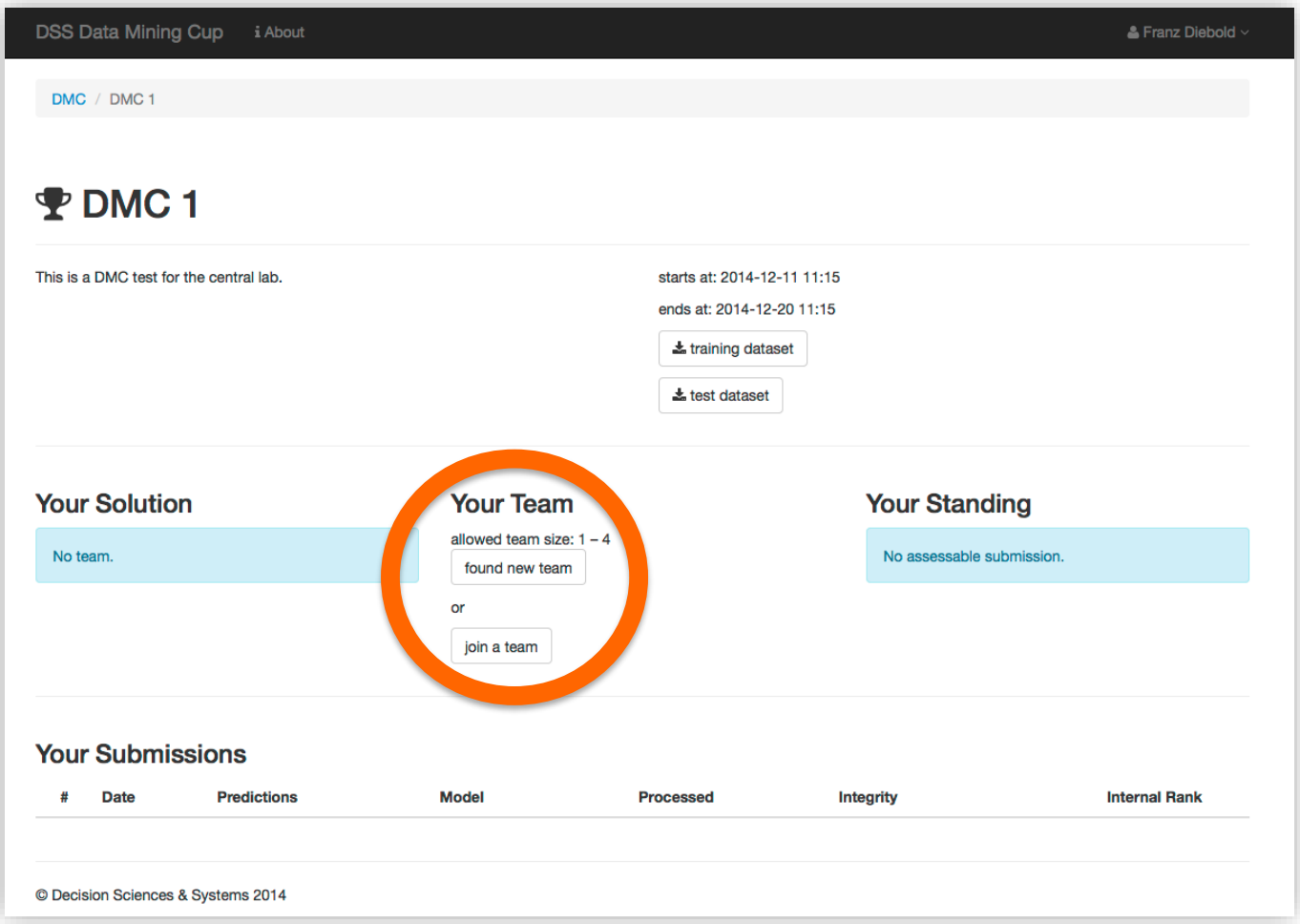
1. Build Team in DMC Manager

Choose the DMC instance in the DMC Manager



1. Build Team in DMC Manager

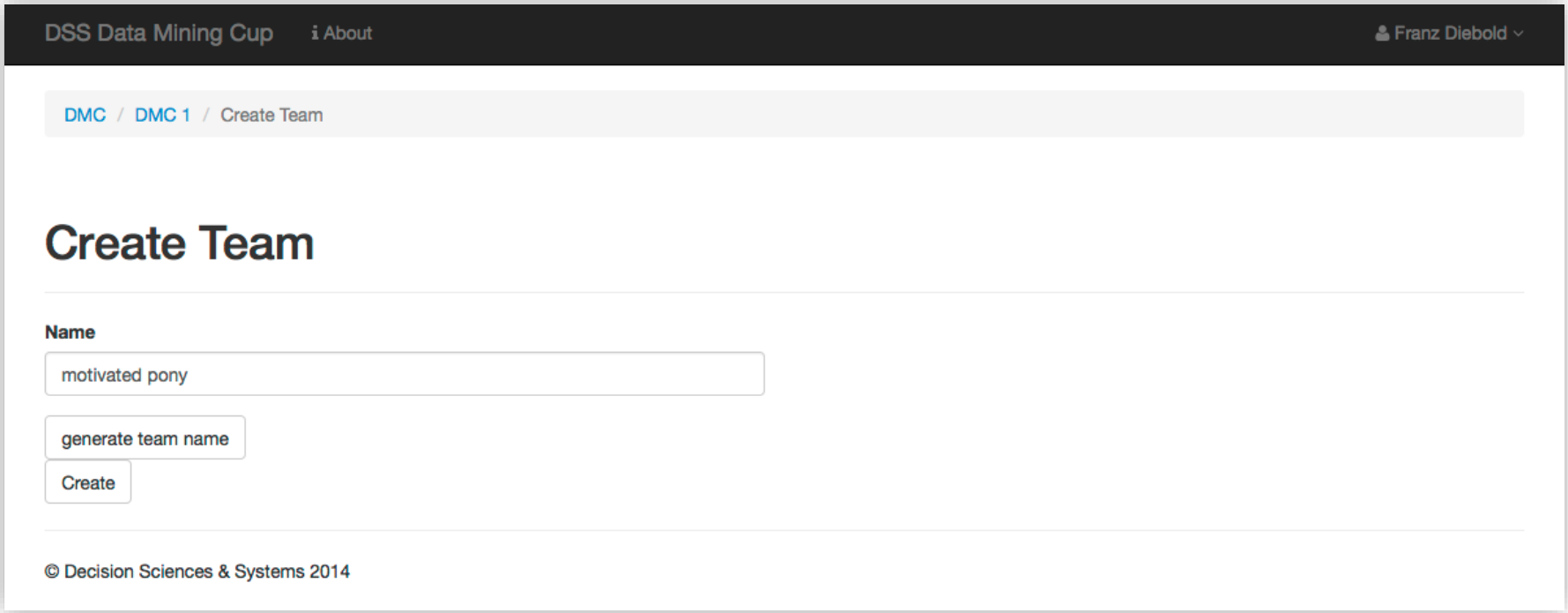
Found new team or join an existing team



1. Build Team in DMC Manager

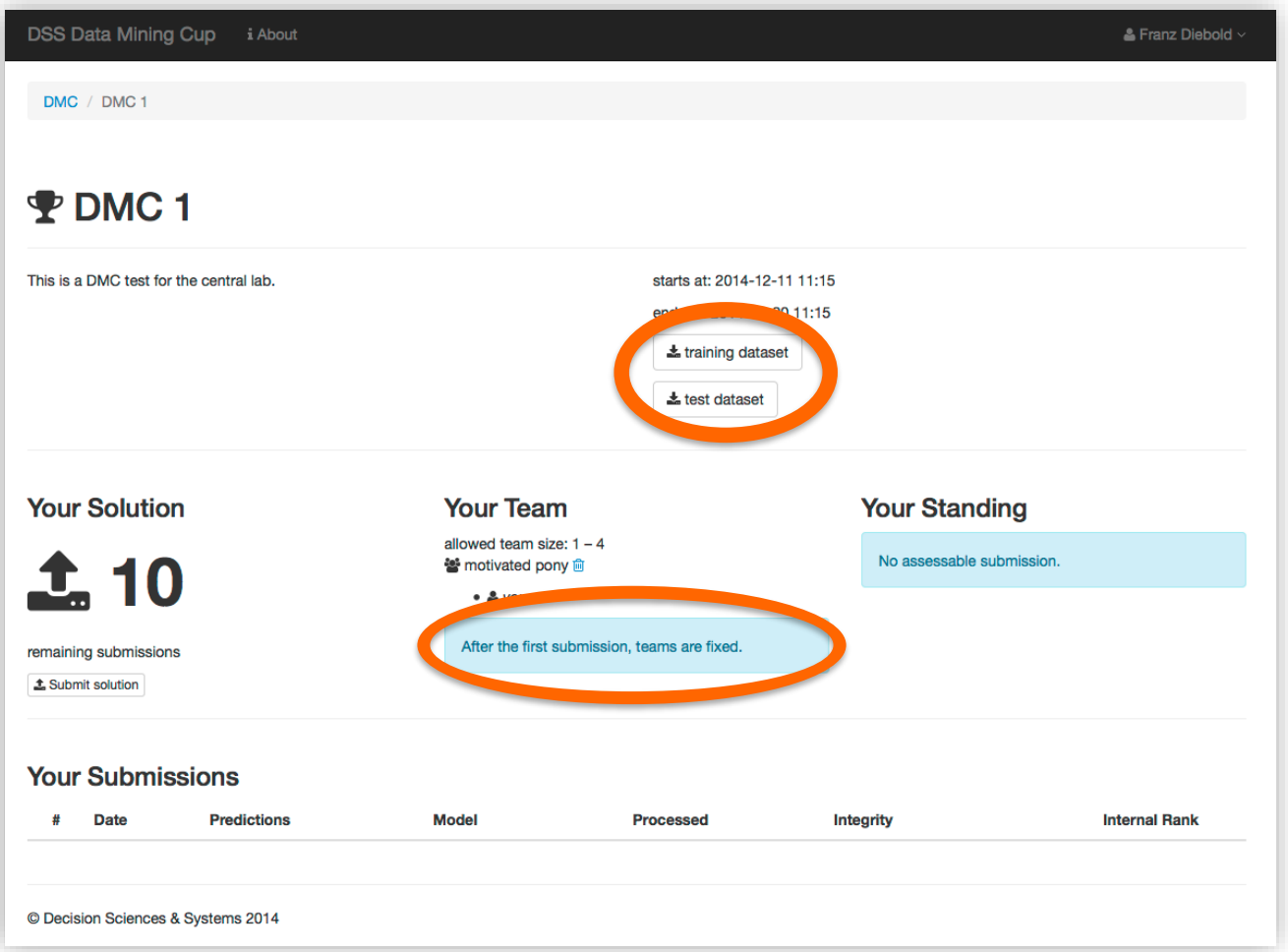
Creating a new team

- Team size: 1-4 members



2. Load & Explore the Data Set

Download the training and test datasets from the DMC Manager

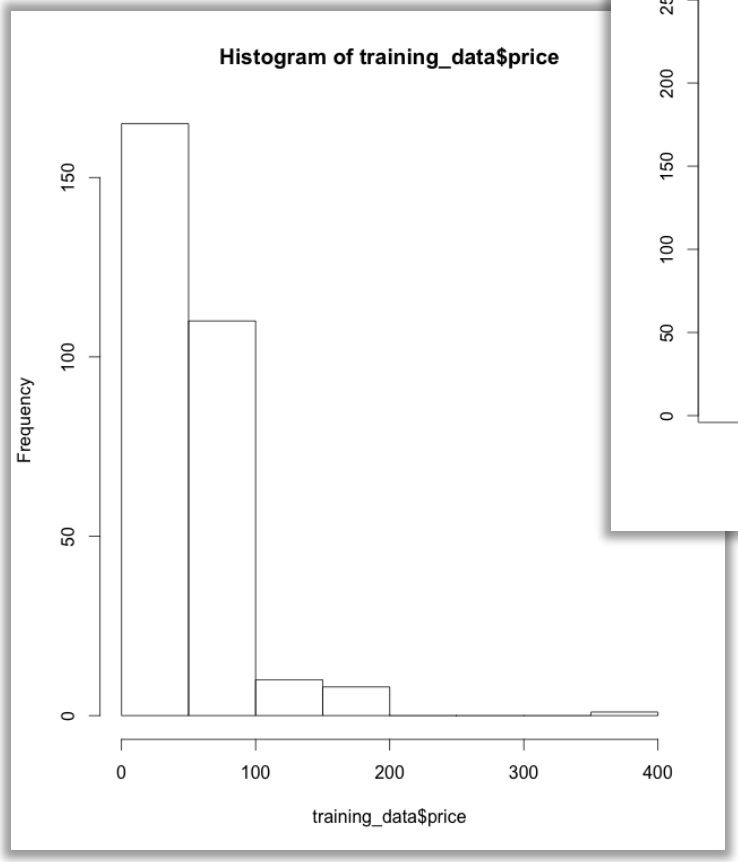


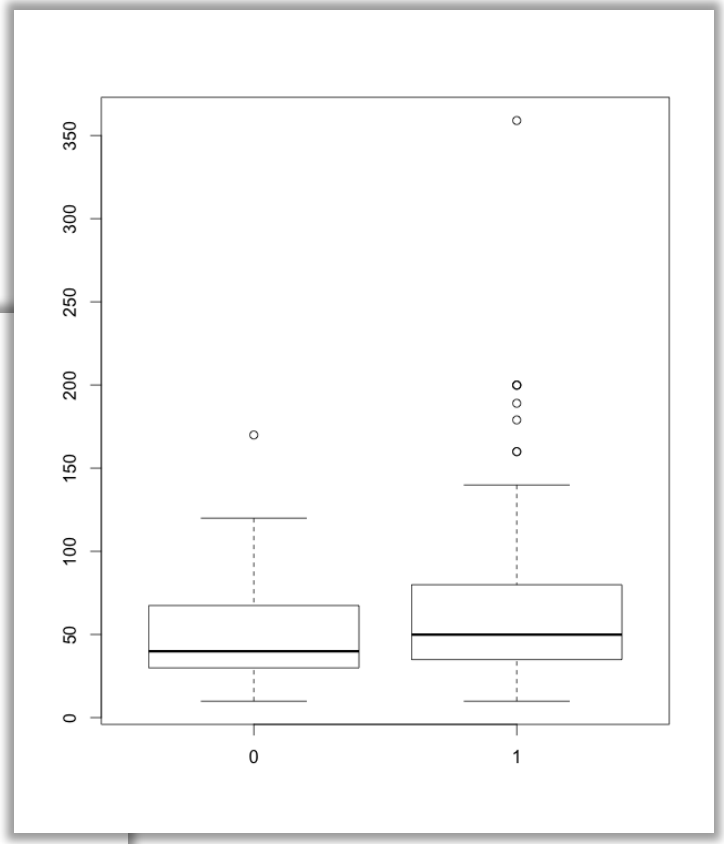
2. Load & Explore the Data Set

Load & Explore in R (compare Tutorial 1)

- Load data sets into R

- Explore the Data Set
  - Get an overview
  - Statistics
  - Plotting





3. Data Preparation

(compare Tutorial 7)

- Possible Data Preparation steps
  - Nominal attributes
  - Ordinal attributes
  - Unified date format
  - Missing values
  - Fix errors and outliers
  - Zero variance and correlation
  - Discretization/binning
  - Feature selection

- ALL changes in both training & test dataset!
- Do NOT DELETE any instances in the test data!

4. Training & Evaluation

Classification Methods

Name	***method* Argument in *train* Function**	**Tuning Parameters**
OneRule	OneR	-
Naïve Bayes	nb	fL, usekernel
Logistic Regression	logreg	treesize, ntrees
Decision Trees	J48	C (pruning factor), M
k-Nearest Neighbors	kknn	kmax, distance, kernel
Ensemble Methods	ada, LogitBoost, logicBag	iter; maxdepth; nu, nIter, nleaves, ntrees

```
> model = train(Class~., data=training, method="J48")
```

More classifiers: http://topepo.github.io/caret/modelList.html

Source: http://topepo.github.io/caret/

4. Training & Evaluation

Classification Methods – Tuning Parameters

- `tuneLength`: number of tuning parameter values
- `tuneGrid`: for specific tuning parameter values
  - data frame, where each row is a tuning parameter setting and each column is a tuning parameter

```
> model = train(Class~., data=training, method="J48",
                tuneGrid=data.frame(C=c(0.1, 0.2, 0.3),M=c(2,2,2))
```

Where to find parameters?

http://topepo.github.io/caret/train-models-by-tag.html

Or in R:

```
> getModelInfo()$J48$parameters
```

4. Training & Evaluation

Metrics

Name	*metric* in *train* Function	Description
Accuracy	Accuracy	*=(tp + tn) / (tp + fp + tn + fn)*
Kappa	Kappa	see below
ROC Curve	ROC	area under the ROC curve

```
> model = train(Class~., data=training, method="J48",
        metric="Kappa")
```

Kappa

- Ratio, which compares a classification method with a random classifier
  - < 0: worse than random classifier
  - > 0: better than random classifier

Source: http://topepo.github.io/caret/

4. Training & Evaluation

Resampling Methods

Name	***method* Argument in *trainControl* Function**
Bootstrapping (Holdout method, default)	boot
Repeated K-fold Cross Validation	repeatedcv
Leave-one-out	LOOCV

```
> # 2 x repeated 3-fold cross validation
> fitCtrl = trainControl(method="repeatedcv", number=3, repeats=2)

> model = train(Class~., data=training, method="J48",
                trControl=fitCtrl)
```

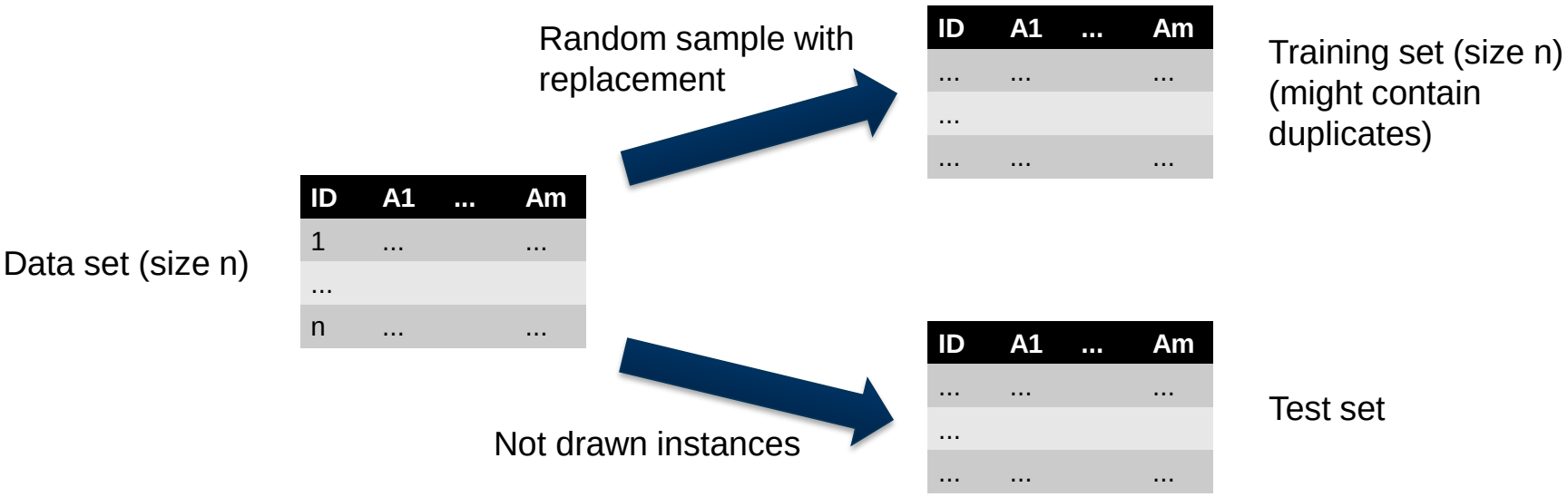
Source: http://topepo.github.io/caret/

4. Training & Evaluation

Bootstrapping

Bootstrapping

- Resampling method



4. Training & Evaluation

Comparing the models

- Can compare several trained models
- The models should be using the same resampling

```
> res = resamples(list(dt = model_dt, nb = model_nb))
> summary(res)

…
Accuracy
     Min. 1st Qu. Median   Mean 3rd Qu.   Max. NA's
dt 0.4457  0.4810 0.4946 0.4910  0.5041 0.5275    0
nb 0.5000  0.5163 0.5246 0.5192  0.5275 0.5275    0
```

5. Predict Classes in Test Data

- Use the trained model to predict the classes in the test dataset.

```
> prediction_classes = predict.train(object=model,
      newdata=test_data, na.action=na.pass)
> predictions = data.frame(id=test_data$id,
      prediction=prediction_classes)
```

6. Export the Predictions

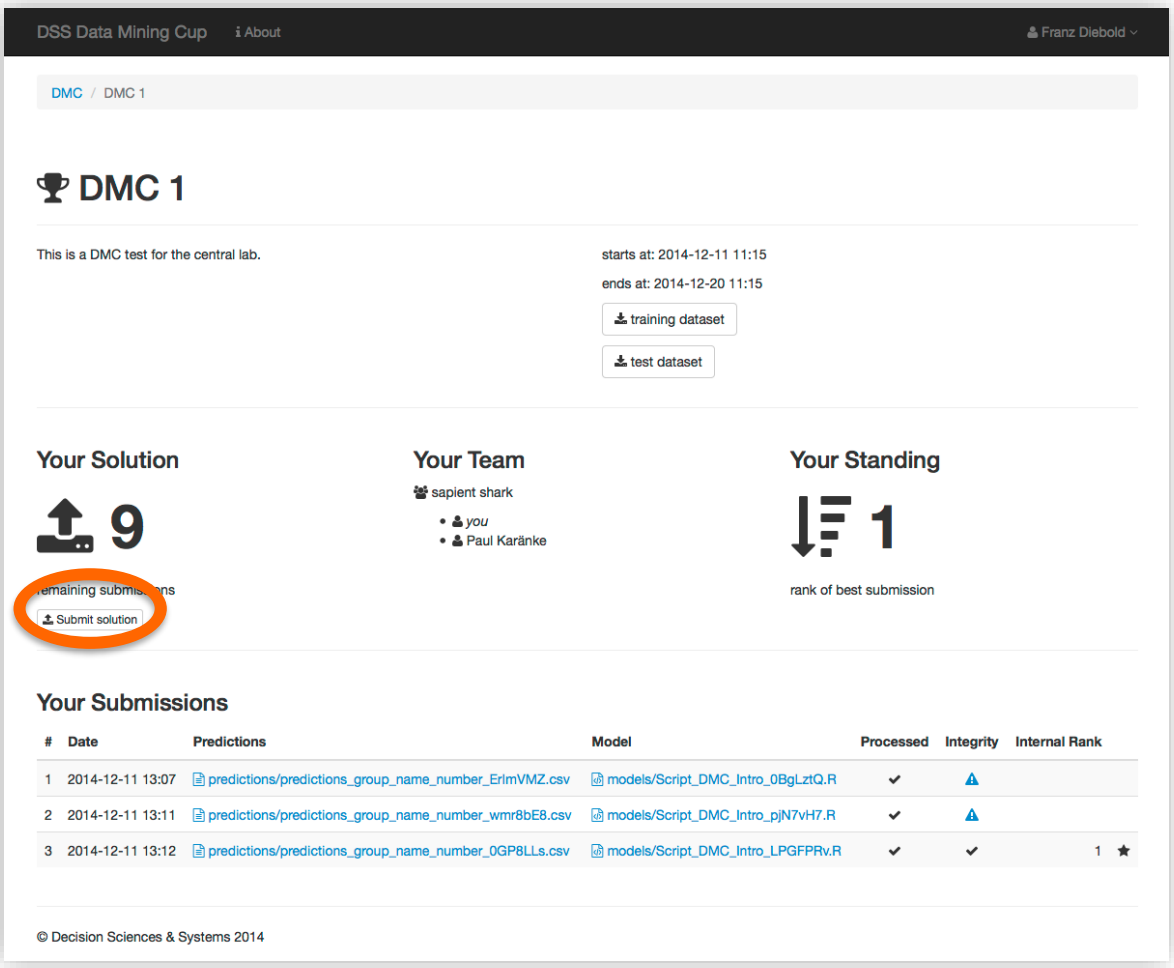
- Export predictions into csv-file
  - Format: id, prediction
  - Must contain all instances of the original test dataset

```
> write.csv(predictions, file="predictions_group_name_number.csv",
        row.names=FALSE)
```

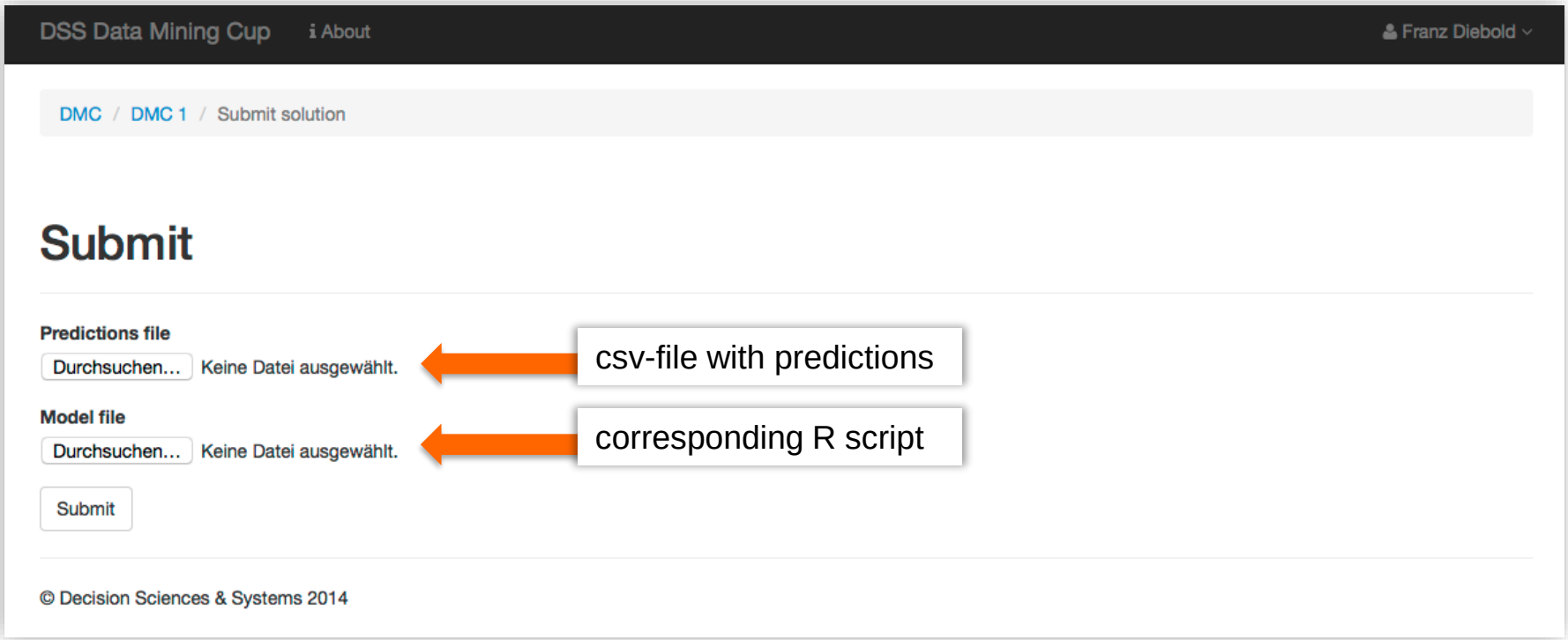
predictions_group_name_number.csv

```
"id","prediction"
130200,"1"
394720,"0"
87847,"1"
228637,"1"
189299,"0”
262991,"1”
…
```

7. Upload the Predictions and the Corresponding R Script on DMC Manager



7. Upload the Predictions and the Corresponding R Script on DMC Manager



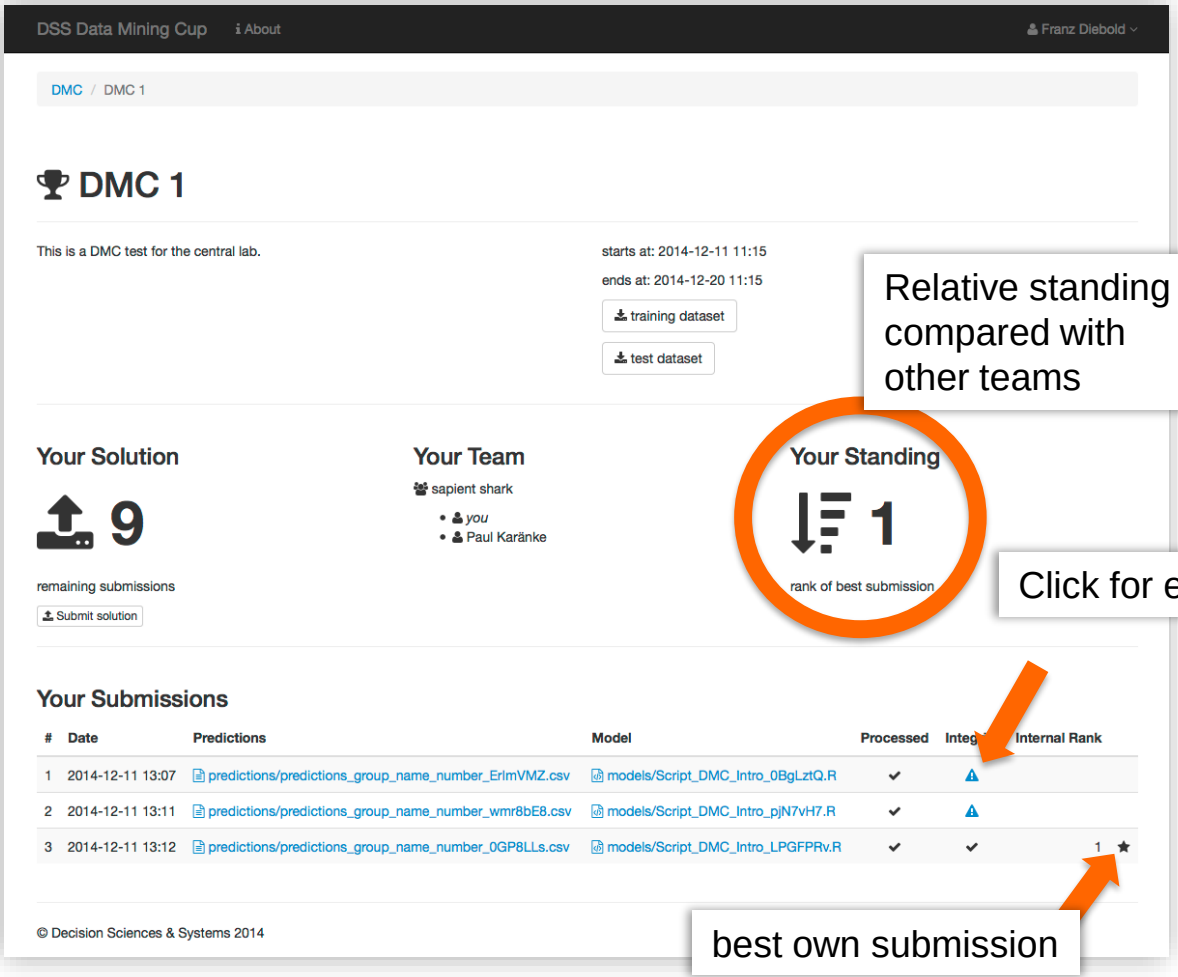
7. Upload the Predictions and the Corresponding R Script on DMC Manager

Submissions & Possible Errors

- Maximum number of submission: 10 (valid submissions)
  - Best submission counts

- Possible errors
  - Wrong column names
  - Unknown IDs (if not in Test Data)
  - Missing IDs (if in Test Data but not in Predictions)
  - Wrong file format
  - …

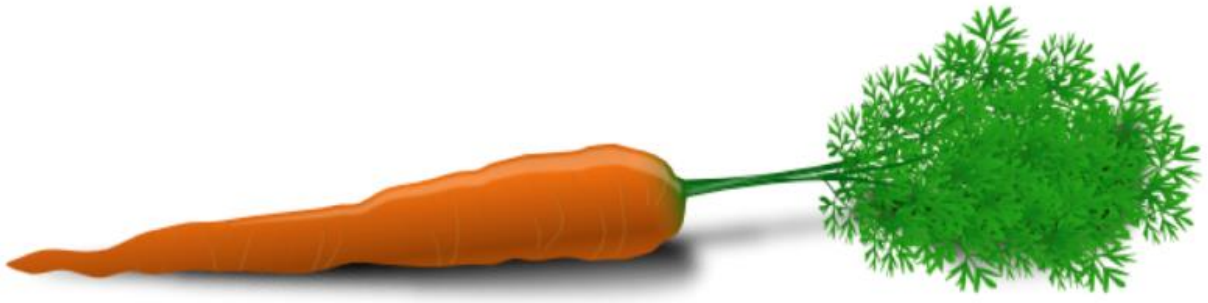
7. Upload the Predictions and the Corresponding R Script on DMC Manager



Questions?

Information about the „caret package“

http://topepo.github.io/caret/



Example dataset raw_data_large

Data

- History of purchase of an online shop
- Both information about good and customer

Task

- Predict if there would be a return

Column name	Description	Range of values	Missing values
ID	Order id	Natural number	No
od	Order date	Date	No
dd	Delivery date	Date	Yes
size	Item size	String	No
price	Price of item	Positive real number	No
tax	Tax	Positive real number	No
a6	Salutation	String	No
a7	Date of birth	Date	Yes
a8	State	String	No
a9	Return shipment	{0,1}	No

DMC Dataset

Data

- History of availability of electric vehicle charging stations

Task

- Predict if a charging station is “in use” (1) or available (0)

Sponsor

- BMW
- **Best team will win Apple Watches!**

Column name	Description	Range of values	Missing values
Id	Ping id	Natural number	No
portNumber	Port at charging station	Natural number	No
TimeStamp	Time of ping	Date	No
EI65_GEO_ID	Station id	Natural number	No
HOUSEHOLD_COUNT	Number of households in neighborhood	Natural number	Yes
TYP1_COUNT	Socket type	Natural number	No
FREECHARGE	No cost for charging	String	Yes
ADDR_LATITUDE	Location	Number	No
ADDR_LONGITUDE	Location	Number	No
ADDR_MUNICIPALITY	City	String	Yes
ADDR_POSTALCODE	Postal code	Number	No
ADDR_REGION	State	String	No
ADDR_STREET	Street	String	No
LAST_MODIFIED	Update of station details	Date	No
PREFERRED_PARTNER	Partner of charge now	String	No
VALIDATION_LAST_MODIFIED	Last time a BMW customer charged there	Date	No
IS_LSC_VALIDATED	Validation technique	String	No
Status	Availability of station	{0,1}, 0 for available	No

Practice and Actual DMC

- Practice DMC is only for getting used to the system and some additional practice if needed
- DMC (WS17/18) is the actual DMC, relevant for the Bonus