Codebook

Coleen Smith

12 April 2019

Contents

Project Description 1
Creating the tidy data set (final_tidyframe) 1
Description of the variables in the tiny_data.txt file 2
An Alternative Codebook - dataMaid 8
Reources 8

Project Description

This assignment will create one R script called run_analysis.R that does the following.

```
1. Merges the training and the test sets to create one data set.
2. Extracts only the measurements on the mean and standard deviation for each measurement.
3. Uses descriptive activity names to name the activities in the data set.
4. Appropriately labels the data set with descriptive variable names.
5. From the data set in step 4, creates a second, independent tidy data set with the average of each va
```

Notes on the original (raw) data

The data for this project came from the Human Activity Recognition Using Smartphones Data Set that was "built from the recordings of 30 subjects performing activities of daily living (ADL) while carrying a waist-mounted smartphone with embedded inertial sensors." From: Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra and Jorge L. Reyes-Ortiz. A Public Domain Data set for Human Activity Recognition Using Smartphones. 21th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2013. Bruges, Belgium 24-26 April 2013.

Full description
Link to data

Creating the tidy data set (final_tidyframe)

For this assignment, I merged data files from the original UCI Data Set containing, information about subject, activity and a set of 561 features measurements (Part 1). The mean and standard deviation features were extracted (Part2). Activity numbers were replaced by more descriptive activity names (Part 3). Variable names were modified in Part 4. Dashes were converted to underscores and parentheses were removed (part 4). The final product was a tidy data set (final_tidyframe) that represented the average of each variable for each combination of activity and subject (Part 5).

In his paper, "Tidy Data," Wickham discusses the order in which variables should appear in a data set:

> "Fixed variables describe the experimental design and are known in advance... Measured variables are what we actually measure in the study. Fixed variables should come first, followed by measured variables, each ordered so that related variables are contiguous."

By this standard, subject and activityName would be fixed variables and should appear contiguously and first. Measured variables would be the items described by the feature measurements included in the original data set.

For a detailed description of the steps to complete the assignment, please review the assignment Readme.

```
    final_tidyframe <- read.table("final_tidyframe.txt", header = TRUE, sep = ",")
```

Each observation in final_tidyframe includes
- An identifier of the subject who carried out the experiment
- The activity name - the 79 averages of the extracted mean and standard deviation for time and frequency variables

Description of the variables in the tiny_data.txt file

Dimensions: 180 observations of 81 variables.
Variables:

```
##  [1] "subject"                       "activityName"
##  [3] "tBodyAcc_mean_X"               "tBodyAcc_mean_Y"
##  [5] "tBodyAcc_mean_Z"               "tBodyAcc_std_X"
##  [7] "tBodyAcc_std_Y"                "tBodyAcc_std_Z"
##  [9] "tGravityAcc_mean_X"            "tGravityAcc_mean_Y"
## [11] "tGravityAcc_mean_Z"            "tGravityAcc_std_X"
## [13] "tGravityAcc_std_Y"             "tGravityAcc_std_Z"
## [15] "tBodyAccJerk_mean_X"           "tBodyAccJerk_mean_Y"
## [17] "tBodyAccJerk_mean_Z"           "tBodyAccJerk_std_X"
## [19] "tBodyAccJerk_std_Y"            "tBodyAccJerk_std_Z"
## [21] "tBodyGyro_mean_X"              "tBodyGyro_mean_Y"
## [23] "tBodyGyro_mean_Z"              "tBodyGyro_std_X"
## [25] "tBodyGyro_std_Y"               "tBodyGyro_std_Z"
## [27] "tBodyGyroJerk_mean_X"          "tBodyGyroJerk_mean_Y"
## [29] "tBodyGyroJerk_mean_Z"          "tBodyGyroJerk_std_X"
## [31] "tBodyGyroJerk_std_Y"           "tBodyGyroJerk_std_Z"
## [33] "tBodyAccMag_mean"              "tBodyAccMag_std"
## [35] "tGravityAccMag_mean"           "tGravityAccMag_std"
## [37] "tBodyAccJerkMag_mean"          "tBodyAccJerkMag_std"
## [39] "tBodyGyroMag_mean"             "tBodyGyroMag_std"
## [41] "tBodyGyroJerkMag_mean"         "tBodyGyroJerkMag_std"
## [43] "fBodyAcc_mean_X"               "fBodyAcc_mean_Y"
## [45] "fBodyAcc_mean_Z"               "fBodyAcc_std_X"
## [47] "fBodyAcc_std_Y"                "fBodyAcc_std_Z"
## [49] "fBodyAcc_meanFreq_X"           "fBodyAcc_meanFreq_Y"
## [51] "fBodyAcc_meanFreq_Z"           "fBodyAccJerk_mean_X"
## [53] "fBodyAccJerk_mean_Y"           "fBodyAccJerk_mean_Z"
## [55] "fBodyAccJerk_std_X"            "fBodyAccJerk_std_Y"
## [57] "fBodyAccJerk_std_Z"            "fBodyAccJerk_meanFreq_X"
## [59] "fBodyAccJerk_meanFreq_Y"       "fBodyAccJerk_meanFreq_Z"
## [61] "fBodyGyro_mean_X"              "fBodyGyro_mean_Y"
## [63] "fBodyGyro_mean_Z"              "fBodyGyro_std_X"
## [65] "fBodyGyro_std_Y"               "fBodyGyro_std_Z"
## [67] "fBodyGyro_meanFreq_X"          "fBodyGyro_meanFreq_Y"
## [69] "fBodyGyro_meanFreq_Z"          "fBodyAccMag_mean"
## [71] "fBodyAccMag_std"               "fBodyAccMag_meanFreq"
## [73] "fBodyBodyAccJerkMag_mean"      "fBodyBodyAccJerkMag_std"
```

```
## [75] "fBodyBodyAccJerkMag_meanFreq"  "fBodyBodyGyroMag_mean"
## [77] "fBodyBodyGyroMag_std"          "fBodyBodyGyroMag_meanFreq"
## [79] "fBodyBodyGyroJerkMag_mean"     "fBodyBodyGyroJerkMag_std"
## [81] "fBodyBodyGyroJerkMag_meanFreq"
```

Fixed Variables

1. subject

Identifies the subject who performed the activity for each sample.

Class: Integer
Values: Range from 1 -30
Source: UCI HAR Data Set subject_test.txt and subject_train.txt

2. activityName

Activity name. Assigned in Part 4, using the dictionary of activity numbers and names provided by activity_labels.txt. The original activity number was replaced by the more descriptive activity name.

Class: Factor with 6 levels
Values: LAYING, SITTING, STANDING, WALKING, WALKING_DOWNSTAIRS, WALKING_UPSTAIRS
Source: UCI HAR Data Set activity_labels.txt

```
    activity_labels <- read.table("UCI_HAR_Dataset/activity_labels.txt", header = FALSE)
    print(activity_labels)
```

```
##   V1                 V2
## 1  1            WALKING
## 2  2   WALKING_UPSTAIRS
## 3  3 WALKING_DOWNSTAIRS
## 4  4            SITTING
## 5  5           STANDING
## 6  6             LAYING
```

Measured Variables

The remaining variables in column 3-81 are the average extracted mean and standard deviations for the time and frequency variables: mean (mean), mean (mean frequency) and the mean (standard deviation).

Class: Numeric
Values: {-1, 1]. Features in the original file were normalized and bounded within [-1,1].
Units of Measurement:
- Acc = Acceleration signal from the smartphone accelerometer X, Y & Z axis measured in standard gravity units 'g'. Acceleration is measured in meters/second^2. Gravity is 9.8 meters/second^2.
- Gyro = Angular velocity vector measured by the gyroscope in radians/second. **Source:** UCI HAR Data Set X_test.txt. X_train.txt

Additional Notes Summarized from the original UCI HAR README.txt and features_info.txt about the features selected and naming schema for this data set:

- Domain indicated by a "t" for time or an "f" for frequency
- Feature measurement derived from the accelerometer and gyroscope 3-axial raw signals tAcc-XYZ and tGyro-XYZ.

- The acceleration signal was then separated into body and gravity acceleration signals (tBodyAcc-XYZ and tGravityAcc-XYZ)
- The body linear acceleration and angular velocity were derived in time to obtain Jerk signals (tBodyAccJerk-XYZ and tBodyGyroJerk-XYZ).
- The magnitude of these three-dimensional signals were calculated using the Euclidean norm (tBodyAccMag, tGravityAccMag, tBodyAccJerkMag, tBodyGyroMag, tBodyGyroJerkMag).
- A Fast Fourier Transform (FFT) was applied to some of these signals producing fBodyAcc-XYZ, fBodyAccJerk-XYZ, fBodyGyro-XYZ, fBodyAccJerkMag, fBodyGyroMag, fBodyGyroJerkMag.
- '_XYZ' is used to denote 3-axial signals in the X, Y and Z directions.

3. tBodyAcc__mean__X

4. tBodyAcc__mean__Y

5. tBodyAcc__mean__Z

6. tBodyAcc__std__X

7. tBodyAcc__std__Y

8. tBodyAcc__std__Z

9. tGravityAcc__mean__X

10. tGravityAcc__mean__Y

11. tGravityAcc__mean__Z

12. tGravityAcc__std__X

13. tGravityAcc__std__Y

14. tGravityAcc__std__Z

15. tBodyAccJerk__mean__X

16. tBodyAccJerk__mean__Y

17. tBodyAccJerk__mean__Z

18. tBodyAccJerk__std__X

19. **tBodyAccJerk__std__Y**

20. **tBodyAccJerk__std__Z**

21. **tBodyGyro__mean__X**

22. **tBodyGyro__mean__Y**

23. **tBodyGyro__mean__Z**

24. **tBodyGyro__std__X**

25. **tBodyGyro__std__Y**

26. **tBodyGyro__std__Z**

27. **tBodyGyroJerk__mean__X**

28. **tBodyGyroJerk__mean__Y**

29. **tBodyGyroJerk__mean__Z**

30. **tBodyGyroJerk__std__X**

31. **tBodyGyroJerk__std__Y**

32. **tBodyGyroJerk__std__Z**

33. **tBodyAccMag__mean**

34. **tBodyAccMag__std**

35. **tGravityAccMag__mean**

36. **tGravityAccMag__std**

37. **tBodyAccJerkMag__mean**

38. **tBodyAccJerkMag__std**

39. **tBodyGyroMag__mean**

40. **tBodyGyroMag_std**

41. **tBodyGyroJerkMag_mean**

42. **tBodyGyroJerkMag_std**

43. **fBodyAcc_mean_X**

44. **fBodyAcc_mean_Y**

45. **fBodyAcc_mean_Z**

46. **fBodyAcc_std_X**

47. **fBodyAcc_std_Y**

48. **fBodyAcc_std_Z**

49. **fBodyAcc_meanFreq_X**

50. **fBodyAcc_meanFreq_Y**

51. **fBodyAcc_meanFreq_Z**

52. **fBodyAccJerk_mean_X**

53. **fBodyAccJerk_mean_Y**

54. **fBodyAccJerk_mean_Z**

55. **fBodyAccJerk_std_X**

56. **fBodyAccJerk_std_Y**

57. **fBodyAccJerk_std_Z**

58. **fBodyAccJerk_meanFreq_X**

59. **fBodyAccJerk_meanFreq_Y**

60. **fBodyAccJerk_meanFreq_Z**

61. **fBodyGyro__mean__X**

62. **fBodyGyro__mean__Y**

63. **fBodyGyro__mean__Z**

64. **fBodyGyro__std__X**

65. **fBodyGyro__std__Y**

66. **fBodyGyro__std__Z**

67. **fBodyGyro__meanFreq__X**

68. **fBodyGyro__meanFreq__Y**

69. **fBodyGyro__meanFreq__Z**

70. **fBodyAccMag__mean**

71. **fBodyAccMag__std**

72. **fBodyAccMag__meanFreq**

73. **fBodyBodyAccJerkMag__mean**

74. **fBodyBodyAccJerkMag__std**

75. **fBodyBodyAccJerkMag__meanFreq**

76. **fBodyBodyGyroMag__mean**

77. **fBodyBodyGyroMag__std**

78. **fBodyBodyGyroMag__meanFreq**

79. **fBodyBodyGyroJerkMag__mean**

80. **fBodyBodyGyroJerkMag__std**

81. **fBodyBodyGyroJerkMag__meanFreq**

An Alternative Codebook - dataMaid

While researching codebooks, I found an interesting package called dataMaid. I had originally book marked it for later, but then ran across it again in one of the class discussion forums. In addition to information similar to summary(), dataMaid produces graphical outputs. I can see why it would be such a useful tool for cleaning data: detecting outliers, missing data and incorrectly enetered data.

I include the dataMaid generated codebook for final_tidyframe in the repository as a supplement and for fun. It does create an R Markdown file which could be edited, but edits would have to wait for the very end so they wouldn't get overwritten during the project iterations. More useful would be to look into the attribute options available to add to the data set itself.

For more information on dataMaid see the resources below and https://CRAN.R-project.org/package=dataMaid

Reources

Henry, Lionel, and Hadley Wickham. "Tidy Evaluation." Tidy Evaluation, tidyeval.tidyverse.org/index.html.

Olson,Molly "R-Ladies: Introduction to Data Cleaning with dataMaid," 19 March 2018, http://biostat.mc.vanderbilt.edu/wiki/pub/Main/MollyOlson/R-ladies-datamaid.html

sandsynligvis.dk, "Generating codebooks in R," 2 March 2018, By https://www.r-bloggers.com/generating-codebooks-in-r/

Schut, Joris, Codebook Template, 22 March 2015, https://gist.github.com/JorisSchut/dbc1fc0402f28cad9b41

Thoughtfulbloke. "Getting and Cleaning the Assignment." Thoughtfulbloke Aka David Hood, 26 Jan. 2016, thoughtfulbloke.wordpress.com/2015/09/09/getting-and-cleaning-the-assignment/.