Fall 2017 - W205 – Storing and Retrieving Data
Week 10 Live Class Session Agenda
Kevin R. Crook

Schedule

- Lab 7 – due Tuesday, 11/14/2017 at 11:59 pm
  - Data Visualization – Introduction to using Tableau with Hive
- Lab 8 – due Tuesday, 11/28/2017 at 11:59 pm
  - Data Exploration / Data Cleansing – OpenRefine – Levenshtein Distances Calculations
- Asynchronous for next week
  - Unit 11 – Graph Models and Analysis
- **No class**
  - Tuesday 11/21/2017 (to keep all 3 sections in synch)
    - Special Extended Office Hours Tuesday night
    - Please slack me to let me know what time you will be there – I'll send you the link to the room via slack
  - Thursday 11/23/2017 – Thanksgiving Day

(next page)

Project

- Milestones
  - Progress Report
    - In class
      - Thursday classes: 11/16/2017
      - Tuesday class: 11/28/2017
    - 10 minute presentation per project team
    - Nothing formal to turn in – please disregard if the official instructions say to turn in something
  - Final Presentations
    - In class
      - Thursday classes: 12/14/2017
      - Tuesday class: 12/19/2017
    - 20 minute presentation per project team
    - All materials must be checked into GitHub repo prior to class time
  - Weeks in which we do not have a formal milestone – we will spend 3 to 5 minutes going round robin through the project teams
- Focus
  - Covering 2 or 3 of the V's
    - Volume, Variety, Velocity
  - Steel Thread
    - Minimal thread working end to end as soon as you can
    - Add to functionality as incrementally as possible
  - Scale Out
    - Decide how much scale out you plan to build in your prototype
    - Balance
      - Build as much scale out as you can
      - But, don't jeopardize having a working prototype at the end of the semester
    - Address any scale out that you didn't build
      - Show a path to get there
    - No Scale Out
      - Storage layer – PostgreSQL, etc.
      - Processing layer – SQL, Python, etc.
      - Machine Learning – Python with Scikit-learn, R, etc.
      - Streaming media – Python single threaded API, etc.
      - Data Visualization – freebee as Tableau has a scale out server

        - Full scale out
            - Storage layer – Hive, Redshift, etc.
            - Processing layer – Spark, Hadoop MapReduce, etc.
            - Machine Learning – Spark MLlib, Mahout, etc.
            - Streaming Media – Storm, Heron, Spark Streaming, Kafka, etc.
            - Data Visualization – freebee as Tableau has a scale out server!
    - Machine Learning
        - Not required, but most teams have at least one member who has experience
        - Steel thread – predictions can be random selection, median, mean, etc.
        - Simple algorithms and enhance time permitting
    - Data Visualizations
        - Easy to add with Tableau
- Tips
    - Instructor does not have to run
        - In exercise 1 and exercise 2 the instructor has to be able to clone your code from GitHub and run it
        - For the project, this is not the case
            - You need to just demonstrate the project running
                - Can use a video if you want and use the class presentation to focus on other aspects of your project
                    - Screencast-o-matic – I use for my videos
                    - Camtasia – professional but expensive
        - If you need to run piecemeal in several instances this is fine
            - Example: storm in one instance, hive in another instance, scikit-learn in another instance, Tableau on laptop
            - You just need to prove out the architecture in this prototype, a few manual steps is ok as long as the major pieces run scripted and you can demonstrate this
    - Multiple users in the same instance
        - Give the key out to everyone and everyone can connect
        - Coordinate starting and stopping the instance with proper shutdowns
        - Leaving instances running is easiest, but can get expensive

(next page)

- If you want to run scripts in your instance without having to stay logged in (or risking a short drop in the network connection logging you out)
  - nohup ./my_bash_script.sh >nohup.out 2>nohup.err &
    - nohup means "no hang up" which means the script keeps running even if you logout or if your connection is lost
    - >nohup.out will redirect standard output to a file nohup.out (or pick another name) so you don't lose output
    - 2>nohup.err will redirect standard error to a file nohup.err (or pick another name) so you don't lose error message
    - & means to run it in the background
  - ps -ef | grep -i my_bash_script
    - use this command when you log back in to see if it's still running and also look at the output files above
  - cron is a scheduling tool on linux, it's easy to use as far as scheduling goes, but environment setup can be a beast as it does not use the login environment you have to set everything
- scikit-learn tips
  - incremental learning
    - some of the algorithms support this
      - warm_start parameter controls this
      - call fit() method multiple times with warm_start=True
    - out-of-core learning
      - reads instances one at a time so the entire training set does not have to be loaded all at once
      - great for when you need to run through a lot of training data and it won't all fit at same time
  - saving object to files using pickle
    - core Python has a module called pickle that can save off any object to a file (or to serial memory)
    - SERDE – serialize / de-serialize
    - Models in scikit-learn are objects, so you can use pickle to save off scikit-learn models
    - Great for when learning takes a long time
    - Great combination with the incremental learning
      - Run, learn, save, stop instance, restart instance, run again, restore from pickle file, continue learning, etc.

(next page)

- Spark tips
  - Processing – using RDDs, lambdas, and transforms (map(), flatmap(), etc.)
    - This is a great opportunity to get experience with Massively Parallel Processing (MPP), Big Data Architecture, and Spark.
      - Good choice since you already know Python and it's an extension of your Python knowledge
    - However, if you are new to it, you may want to limit this to a subset of processing, or do the steel thread in core Python and move pieces to Spark time permitting.
  - Machine Learning using MLlib
    - This is a great opportunity to get experience with Machine Learning in a Massively Parallel Processing (MPP) environment, a Big Data Architecture, and Spark.
      - Good choice since you already know Python and it's an extension of your Python knowledge
    - However, MLlib is not easiest product to work with. It's also not as stable as you might be used to with other tools.
      - You may want to first use scikit-learn (with grid searches, pipelines, etc.) to narrow down your algorithm of choice, hyperparameters, etc. then implement it in MLlib.

Today in class

- Project
  - Not a milestone – quick round robin
- Tableau Exercise
  - Go all the way through Tableau – more details than lab 7
  - Decide if we want to do break out groups or do together as a class

(next page)

Tableau Exercise

- Create a new Tableau workbook with a downloaded data set of coffee chain data
    - Download the coffee shop dataset
        - http://kevincrook.com/ucb/data/coffee_chain.xlsx
    - Open Tableau => new workbook => connect to a file Excel => coffee_chain.xlsx
    - Data Source tab => drag the CoffeeChain_Query (table)
    - Upper right => choose the Extract radio button
    - Sheet1 tab => click => it will ask you to save the extracted data in a tde file => save as coffee_chain.tde
    - Save the Tableau workbook: File => save as => coffee_chain.twb
- Create a Bar Chart
    - Rename Sheet 1 to “Profit by Product Type”
    - Measures => Profit => drag to Columns shelf
    - Dimensions => Product Type => drag to Rows shelf
    - Tool bar => hover to find “Swap Rows and Columns” => test and see which way looks best
- Create a Side by Side Bar Chart
    - Profit by Product Type tab => right click => duplicate => rename new sheet to “Profit by Market and Product Type”
    - Dimensions => Market => drag to front of Product Type in Rows shelf
    - Tool bar => hover to find “Swap Rows and Columns” => test and see which way looks best
- Create a Stacked Bar Chart
    - Profit by Market and Product Type tab => right click => duplicate => rename new sheet to “Profit Stacked by Market and Product Type”
    - Market in Rows shelf => drag to the Marks card, Color shelf
    - Tool bar => hover to find “Swap Rows and Columns” => test and see which way looks best
- Create a Time-series Line Chart
    - Tab bar => hover to find “New Worksheet” => click to create a new worksheet and rename it “Profit over Time”
    - Measures => Profit => drag to Rows shelf
    - Dimensions => Date => drag to Columns shelf => click the drop down to change it from YEAR to MONTH(Date) (for this one, pick the second MONTH in the drop down that has the year with it)
    - Dimensions => Market => drag to the Marks card, Color shelf

- Create a Monthly Year-over-Year Comparison Time-series Line Chart
    - Profit over Time tab => right click => duplicate => rename new sheet to "Monthly Year-over-Year"
    - Dimensions => Date = drag to the Marks card, Color shelf to replace the Market
    - Columns shelf => MONTH(date) => dropdown and change from the second months (with the year) to the first MONTH (without the year)
- Create a Filled Map Geographical Data Visualization (aka Choropleth Map)
    - Tab bar => hover to find "New Worksheet" => click to create a new worksheet and rename it "Profit by State"
    - Dimensions => State => double click (notice how it automatically puts Longitude in the Columns shelf and Latitude in the Rows shelf)
    - Measures => Profit => drag to the Marks card, Color shelf
- Create a Symbol Map Geographical Data Visualization (aka Choroplethic Heat Map)
    - Tab bar => hover to find "New Worksheet" => click to create a new worksheet and rename it "Profit by Area Code"
    - Dimensions => Area Code => double click (notice how it automatically puts Longitude in the Columns shelf and Latitude in the Rows shelf)
    - Measures => Sales => drag to the Marks card, Size shelf
    - Measures => Profit => drag to the Marks card, Color shelf
- "Show Me"
    - Tab bar => hover to find "New Worksheet" => click to create a new worksheet and rename it "Show Me"
    - Select multiple Dimensions and Measures and the Show Me drop down will show you the possible data visualizations that can be generated
    - To select multiple in Windows, click on the first one, then while holding down the control key, click on additional ones
    - Try the following combinations:
        - Dimensions
            - Area Code
            - State
        - Measures
            - Profit
        - Show Me
            - go through all of the visualizations one by one and take a look at the ones that can be generated
            - for the ones that cannot be generated, see why (what's needed in terms of dimensions and measures

- Create an Interactive Dashboard
    - Tab bar => hover to find “New Dashboard” => click to create a new worksheet and rename it “Profit Analysis”
    - Sheets => Profit Stacked by Market and Product Type => drag
    - Sheets => Profit over Time => drag to right half tile
    - Sheets => Profit by Location => drag to the left lower tile
    - May want to resize sheets a bit to make them look better
    - Profit Stacked by Market and Product Type => click on the sheet to activate it, title bar should appear => upper right of title bar has a down arrow, funnel or tornado shaped, X => click on the funnel to activate filtering
    - Experiment by clicking on various parts of the stacked bar chart and see what happens to the other charts (remember you can select more than one by holding down the control key while clicking)
- Create a Data Story
    - Tab bar => hover to find “New Story” => click to create a new worksheet and rename it “Coffee Chain Profit Data Story”
    - Take multiple data visualizations and/or Dashboards (or the same Dashboard with multiple filters set) and put story boards to describe what is going on
        - Experiment by making 2 or 3 story boards
    - Data Stories are very popular these days – especially with executives
    - Consider creating a Data Story for your final analysis for your project