Bonus Assignment

15 Bonus Points

General Description:

Different tasks in the software development process like testing, inspections, and reviews are the means by which we control and manage the quality of the software products when we plan, execute, manage, and monitor the software projects. It is common to have outliers in datasets of software metrics, change requests, and quality records, however, outliers in these datasets are often the most interesting metrics that often used to detect potentially useful patterns of hidden problem. The S-curve and Bell-curve are the tools used to track and monitor the healthy progress of the project while executing the software project plan and compare the planned numbers to the actual numbers of tasks executed, effort spent, change requests, issues created, and defects detected. However, analyzing the outlyingness of issues and defects reported with respect to the average expectations might be indicative of hidden and potentially useful patterns to detect problematic areas in the workflow of the software development process lifecycle or the software project plan.

General Instructions:

1. Write a technical term paper that presents and analyzes the experimental results that can help in detecting outliers in the provided dataset of issues that might indicate hidden problems in the given data set of issues for project X

2. Modify the provided python ipynb script to run your experiments to obtain results to use in your analysis

3. You can only get credit for the Bonus assignment if you successfully complete Assignment #1, #2, #3, #4, #5 and the Final Project.

4. Only a tested and runnable ipynb/python script can get a credit; NO partial credit for spaghetti code.

5. Leverage all knowledge and skills you have learned in the assignments and lecture notes when writing your report and discussing the experimental results

6. Submit your Bonus assignment as a single winzip file that has:

    a. the technical paper report
    b. your modified ipynb script that has your code and the output.

7. This is an individual assignment NOT team assignment

8. **Due Date is 4/21/17 by 9:00am**.

Requirements:

The default labels that GitHub provides us with are very primitive and very insufficient for hierarchical indexing and filtering. Your task for this assignment is to modify the provided ipynb script to answer simple and complex queries, chart the data, and provide a report that analyzes the experimental results to detect outliers that might be used to indicate patterns of potentially hidden problems in the given dataset .

1) **Issues Tracking and Report Design**: Provide Python ipynb script that will allow project managers and quality managers to answer and chart all types of queries related to issues/tickets/change requests/modification requests; the following queries are only a sample of queries that your python script must be able to answer and provide the output report/chart
    i. Plot in Bar Chart the total number of issues created every day
    ii. Plot in Bar Chart the total number of issues created every day and originated in Design phase
    iii. Plot in Stacked Bar Chart the total number of issues based on their priorities created for every originating phase
    iv. Plot in Bar Chart the total number of issues closed in every day for every **DetectionPhase** that have labels (**Category:Bug,Priority:Critical, Status:Completed**)
    v. Plot in Pivot Chart the total number of issue status created for every **OriginationPhase** (group by originating phase)
    vi. Plot in Control Chart for the total number of Critical issues created every week and originated in Design phase. Your Control chart must plot/show the UCL (Upper Control Limit) and LCL (Lower Control Limit)

2) **Labeling Scheme**: The Labeling Scheme follows the **key:value** format "**LabelName:*LabelValue***" for every label

3) **Types of Labels**: There are different types of labels, the following is the list of labels used for issues listed in the provided dataset:
   i. **OriginationPhase**: Could have one of the values{Requirements, Design, Coding, Testing, Documentation, Field}
   ii. **DetectionPhase**: Could have one of the values{Requirements, Design, Coding, Testing, Documentation, Field}
   iii. **Priority**: Could have one of the values{Critical, Major, High, Low, Medium}
   iv. **Status**: Could have one of the values{Approved, Rejected, Completed, inProgress, pendingReview}
   v. **Category**: Could have one of the values{Bug, Enhancement, Inquiry}

4) **Implementation**:
   i. Modify the provided ipynb python script to carry out experiments to obtain the results for your analysis
   ii. Use the provided CSV file for the issues dataset to obtain the experimental results
   iii. Use https://pypi.python.org/pypi/pivottablejs/0.1.0 for pivottable/pivotchart
   iv. All queries listed above must be answered
   v. You are not allowed to use any freeware other than the one discussed in class

5) Data Analysis & Statistical Control Process:

Goals – Create the Intelligent Reports to Support the Decision Making process
Are there any useful patterns of outliers in the dataset?
What are the UCL (Upper Control Limit) and LCL (Lower Control Limit) for certain issue metrics?
Can the outliers detect hidden problems in the given dataset?
What is the correlation between outliers and hidden problems in the given dataset?
How to detect if there is problem hidden in the given dataset?
How to detect if certain engineer deliberately creates issues with Priority Critical?
How to detect if certain engineer issues deliberately Rejected?
How to detect if certain origination phase causes majority of the in progress-Critical-Bug issues?
Can you chart the patterns of outliers in the dataset?
Can you create the right pivot stackedbar chart?
How to group multi-levels? Group by Origination phase or Category for example.
What is the correlation between outliers and hidden problems in a given data?
How many Levels of indexing? Should the Priority be displayed in a pivotchart of DetectionPhase for example
What is the avg number of issues opened per DetectionPhase?
What is the avg turn around time per issue (time from the day the issue created till it got closed)?
What is the avg number of rejected issues opened per eningeer?
What is the avg number of critical issues opened per eningeer?
What is the avg number of rejected issues per OriginationPhase?
What is the avg number of critical issues per OriginationPhase?
What is the avg number of created issues per OriginationPhase?
What is the avg number of rejected critical issues per OriginationPhase?
What is the ratio of total number of critical to medium issues per OriginationPhase?
Which month got the maximum number of Critical issues created?
Which week got the minimum number of issues created?

Deliverables:

You are required to submit a SINGLE WinZip file that has the following deliverables are:

1. Python ipynb script Source Code and Output
2. Technical term paper that presents and analyzes the experimental results of the provided data set

Post your bonus assignment as a SINGLE WIN-ZIP file on Blackboard on 4/21/17 by 9:00am.

Please post your Bonus Assignment under the name "CS587 Bonus - Lastname, FirstName". On the very first page write your name and email address.

Dr. Atef Bader

Appendix A:

Issues Dataset. The issues.csv file has the following layout(you could open the CSV in excel or notepad).

Issue #	OriginationPhase	DetectionPhase	Category	Priority	Status	Created_at	Closed_at	Author
1	Requirements	Coding	Bug	Critical	Approved	2/24/2017		Smith
2	Design	Testing	Enhancement	High	Approved	2/25/2017		Roy
3	Requirements	Design	Inquiry	Low	Rejected	2/26/2017	3/7/2017	Linda
4	Testing	Field	Bug	High	Completed	2/27/2017	3/8/2017	Kim
5	Documentation	Field	Enhancement	Major	pendingReview	2/28/2017		James
6	Design	Coding	Inquiry	High	inProgress	3/1/2017		John
7	Coding	Testing	Bug	Low	Completed	3/2/2017	3/11/2017	Tom
8	Testing	Field	Enhancement	Medium	Completed	3/3/2017	3/12/2017	Lindsey
9	Design	Testing	Inquiry	Critical	Approved	3/4/2017		David
10	Requirements	Coding	Bug	High	inProgress	3/5/2017		Michelle
11	Requirements	Design	Inquiry	Low	Rejected	2/26/2017	3/17/2017	Smith
12	Testing	Field	Bug	Medium	Completed	2/27/2017	3/18/2017	Rose
13	Documentation	Field	Enhancement	Major	pendingReview	2/28/2017		Clark
14	Design	Coding	Inquiry	High	inProgress	3/1/2017		John
15	Coding	Testing	Bug	Low	Completed	3/2/2017	3/3/2017	Lisa
16	Design	Coding	Inquiry	High	inProgress	3/3/2017		Lindsey
17	Coding	Testing	Bug	Low	Completed	3/4/2017	3/15/2017	David
18	Testing	Field	Enhancement	Medium	Completed	3/5/2017	3/16/2017	Catherine
19	Design	Testing	Inquiry	Critical	Approved	3/6/2017		Smith
20	Requirements	Coding	Bug	High	inProgress	3/7/2017		Joseph
21	Requirements	Design	Inquiry	Low	Rejected	3/8/2017	3/19/2017	Leslie
22	Testing	Field	Bug	Medium	Completed	3/9/2017	3/10/2017	John
23	Documentation	Field	Inquiry	Major	pendingReview	3/10/2017		Jessica
24	Design	Testing	Enhancement	High	Approved	3/11/2017		Christopher
25	Requirements	Design	Inquiry	Low	Rejected	3/12/2017	3/13/2017	Smith
26	Testing	Field	Bug	Medium	Completed	3/13/2017		Kim

Appendix B:

https://en.wikipedia.org/wiki/Control_chart
https://en.wikipedia.org/wiki/Six_Sigma
http://www.skymark.com/resources/tools/control_charts.asp