Name: _____

MIS 3335 Data Analysis Using Python
Final Exam – Fall 2020 – 200 points total

This is an individual assignment and must be completed separately by each student. Group work or help from other students is not permitted.

The final exam consists entirely of coding exercises in a Jupyter Notebook created by the student. You will read, clean, plot, and analyze the data. The specific instructions are below. Turn your properly named notebook file (.ipynb) at the exam link before the end of the final exam period.

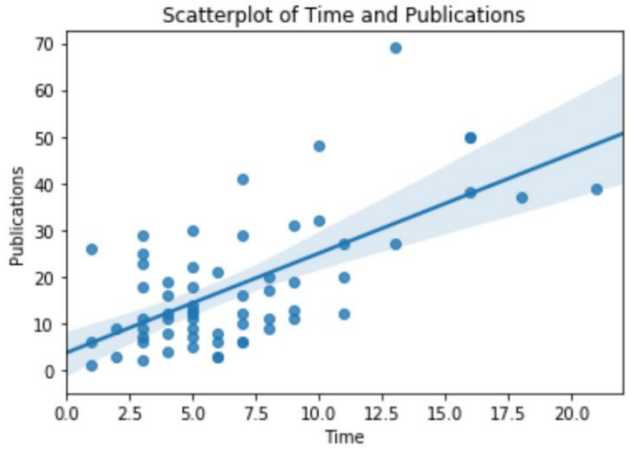
Use the data in the file "`phd_salary_final.csv`" to complete the following tasks. More details about the structure of the data is provided at the end of this document.

Part 1: Data Preparation

1. (30 points) Create a new Jupyter Notebook file using your first initial and last name plus "_final" for its name. Follow the coding standards we have been using this semester as you complete the remaining tasks to produce an easy-to-read notebook that shows your code, documentation, and the results. The standards include the addition of a header cell, using markdown cells for section headers and documentation, commenting your Python code, using meaningful variable names, etc.
2. (30 points) Load the data from the csv file to a dataframe object named "phd." Specify in your read statement that the "id" field in the data be used as the dataframe index. Also, specify which values will be treated as missing values.
3. (20 points) Fill any missing values with the mean of the column in which they appear.
   [If you are unable to do this in your code, manually change missing values in "time" to 7 and in "publications" to 18 so you can complete the remaining steps.]
4. (20 points) Change the tenure column type from "object" to "category."

Part 2: Subsetting, Plotting, and Aggregating

5. (20 points) Use a conditional statement to subset the rows representing faculty members with more than 40 publications. Save these rows in an object named "phd_sub" and display them.
6. (20 points) Use "groupby" to display the totals of all columns for each tenure category.
7. (30 points) Use seaborn to create a scatterplot (using all 62 rows of phd) that looks like this:



Part 3: Regression

8. (30 points) Create a regression model using the "statsmodels formula" library we used previously. Start by using `time`, `publications`, and `citations` to predict `salary`. Based upon those results, explain in a markdown cell what changes need to be made in your model specification to improve the model, and run the modified code in a new code cell. Be sure to display the regression results for each model specification you run.

Submit your Jupyter notebook file at the Blackboard link before the deadline when you are finished.

The Data

The data file `phd_salary_final.csv` consists of the following fields:

Column Name	Description
`id`	A unique identifier for each record in the data.
`time`	The number of years since the subject earned their terminal degree.
`tenure`	The tenure-track faculty rank of the subject: assistant professor (asst), associate professor (assoc), or professor (full).
`publications`	The number of peer-reviewed publications attributed to the subject.
`citations`	The number of times the subject's papers have been cited in other research.
`salary`	The subject's annual salary, in US$.

This is a sample screenshot of the first 15 rows of the data as seen in Excel. The values "-999" and "spam" should be treated as missing values.

	A	B	C	D	E	F
1	id	time	tenure	publication	citations	salary
2	T1291	5	assoc	-999	33	47212
3	T1244	6	assoc	3	26	54511
4	T1300	6	assoc	3	36	41195
5	T1293	5	assoc	5	42	53650
6	T1265	7	assoc	6	18	53740
7	T1248	6	assoc	6	37	47034
8	T1260	7	assoc	6	69	56600
9	T1264	5	assoc	7	35	62895
10	T1258	6	assoc	8	32	54528
11	T1268	8	assoc	9	30	55682
12	T1285	5	assoc	9	47	58632
13	T1255	spam	assoc	10	25	39115
14	T1267	5	assoc	11	60	56596
15	T1302	8	assoc	11	70	47606
16	T1289	5	assoc	12	43	54782