About applied problem sets

We usually have four to six problem sets. They include hands-on work with real data and some theoretical work as well. Some problems are designed to make the class lessons take root by having you see how things work out. Others have you explore things that our limited classroom time does not cover.

Some of the problems in this course have been used (debugged even) before. It is an honor code violation to seek or use solutions from earlier years. Also it defeats the main purpose of these problem sets, which is to learn by doing.

Mechanics

Problem sets are announced in class. You will find the PDF of them in the problem set web page. We will use gradescope for problem set uploads. Details forthcoming.

Late work

We apply a penalty to late work. It is 10% per day late. It needs to be in by midnight on the due date. Each subsequent midnight adds one day of lateness to work not turned in.

It is very common that something comes up around a due date. Illnesses, interviews and sudden travel all occur. We will keep track of your late days. At the end of the quarter, 3 days of lateness are forgiven. We choose the 3 forgiven days to minimize your penalty.

Group work

Some questions are to be done by each student individually. Others may be done in teams of two or three students. For group work, discuss problems with your group. The point of group work is to exchange ideas and learn about the problem, and not divide and conquer with each problem getting done by one person!

For questions done in groups, each student has to write up their own version, and acknowledge their team members. (Just list who they were.)

Writeups

Write your answers up carefully, clearly and legibly. Long passages of text should be typeset (LaTeX is by far the best). If necessary, you can write up
your mathematics by hand, and scan it in, but be sure that it is clear and
dark and large enough to read.

Carefully select what you turn in. Use an appendix to show any code
that needs to be shown. If you’re asked to write some code then put it in
there. Routine calls to plotting functions et cetera don’t need to be handed
in.

Correctness and accuracy are the most important criteria in grading.
But clarity also counts. Typically in applied settings you are writing for a
non-statistician. A better organized and clearer writeup can get a better
grade. A thoughtfully prepared data plot can be worth more than a sloppy
one that comes out of the software by default.

The applied setting is unlike mathematics. Two students might make
different but reasonable assumptions and end up with different answers that
tie for the top grade on a question. Points are taken off for mistakes, or
relatively poor choices, or inadequate reasoning. Points may be added for
answers that are somehow much better than expected.

**TA help**

Please don’t ask the TAs to tell you how to do your problem set. It is fair
to ask them about the statistical concepts underlying the problem. They
can explain the statistical issues to you until you’re able to approach the
problem yourself.

It is also proper to ask the TAs what the problems mean. But it is even
better to ask me. The best time to ask is just before class. Then I might be
able to clear things up for others too.

If your line of questioning seems to be too much like just asking how
to do xyz, then the TAs and I may well decide that we cannot answer you
directly.

On some problems figuring out what you’re supposed to do is actually
what you’re supposed to do. It often happens that choosing is hard but
implementing is easy.

**Errors in the data**

Sometimes there are errors in the data: some numbers are clearly wrong.
I’ve never purposely put in a wrong number, but when I see one in a data
source, I usually don’t fix it. Being alert for possible wrong data is part of
doing statistics. You are expected to detect and remove obvious errors from
the data. If you’re suspicious that something might be wrong with the data,
describe it and give reasons for your suspicions. Then decide whether or not to change the data and give (brief) reasons.

**Errors in the problem sets**

If you find a typographical or other error in the problem set, please tell me as soon as you can. I’ll check it and if necessary send an email to the class about it. Be sure you gave a working email address to Axess.

**Solutions**

Pieces of your work may be clipped into the solutions, with attribution, when it is to be held up as an example of a good (or best) solution. (Your work will not be shown as an example of a mistake.) If necessary, common mistakes will be described some other way.