ATOC5860
Objective Data Analysis

Instructor:
Prof. Jennifer Kay (she/hers) – Call me “Jen”
Jennifer.E.Kay@colorado.edu

Spring 2023
T/Th 2:30-3:45 pm

Today's plan….

1. Discuss prerequisites, expectations, class goals
2. Introductions
3. What are statistics good for (and not good for).
4. Review of basic statistics/Bayes Theorem/Distributions (Barnes 1.1.1-1.1.2)

Syllabus, Grades, Topics Covered in ATOC5860

- Please read the syllabus by next class. Let me know if you have any questions about the course, my expectations of you, topics covered, etc.
- Grade will be based on five homework assignments *turned in on Slack* (50%, complete 5 out of 6), six in-class applications laboratories *turned in on github* (30%), short presentations on analysis of your data completed in homework (10%), and a written review paper (10%). There are no exams.
- Course divided into six units:

(1) General statistics (lectures 1-3, homework #1)

(2) Regression and correlation (lectures 4-5, homework #2)

(3) Matrix methods/EOF/PCA (lectures 6-8, homework #3)

(4) Spectral analysis (lectures 9-10, homework #4)

(5) Filtering (lectures 11-12, homework #5)

(6) Machine Learning (lectures 13-15, homework #6)

How do we communicate outside class??

- ***Use Slack (Not E-mail!).***

You should have gotten an invitation to join our class slack:

https://join.slack.com/t/slack-vtm3976/shared_invite/zt-1nemyancx-moUrZYFGQQ1yQE2_wzLWEQ

https://app.slack.com/client/T04K9A87NGH/C04K9CZKJG1

- ***Office hours in my office SEEC N249:***

Mondays 11am - 1pm

(or by appointment, send me a direct message on Slack)

Where do I find course materials? Google Drive (not Canvas!)

- Syllabus, Schedule, Lecture Notes, Homework, Code, Data, Application Labs… It's all there: https://drive.google.com/drive/folders/1_XPusrYE0URH3Y_vtnQ5zI2tkc8mH_R2?usp=sharing
- Please download and make a copy of all code used in the course. The course google drive (link above) is View Only.
- There is no official textbook for this course. I have provided lecture notes from Prof. Libby Barnes (Colorado State University), which we will follow closely. I have also provided lecture notes from Prof. Dennis Hartmann (University of Washington). ***I recommend reading through the Barnes notes before lecture to identify questions that you might have.***

Prerequisite Knowledge

You are expected to be familiar with the following math:

- algebra
- basic calculus
- basic matrix algebra
- trigonometric functions (sines, cosines)

All programming for the class will by done in Python Jupyter Notebooks.

Plenty of example code will be provided. Add to the repository!!

If you are new to Python, you are not alone. "Jump into the Deep End!"

If you are concerned about pre-requisites, Please contact me immediately.

Analysis software and datasets

- We will be using iPython notebooks in class during lectures and during application laboratories. The code will be provided and you will "tinker" with it to learn and apply the methods we are using in an environment that does not require you to do a lot of coding in class.
- I am not a Python expert, but I do think it is the best language for this class given the recent surge in use in both academic and industry settings. This is a computing reliant class. There are plenty of opportunities for things to go wrong. Be helpful and Be patient…
- Datasets for homework and application laboratories will be provided many formats (e.g., .csv, netcdf). Note that we will not be working with large datasets in this class. Scaling up the methods learned here for much larger datasets is something that I am happy to discuss during office hours.

Why Python?

python™

-Free and Open source!

-General purpose: Can be used for all sorts of programming tasks

-High-level: Less code required for a task than in other languages

-Syntax is focused on readability

-Extensible: Many packages available

My opinion: If you could only pick one language to learn, Python is the most flexible choice right now regardless of your career path.

A Major Drawback of Python

* **Python has many packages that are constantly in development with new versions released frequently**
* **Most packages have dependencies, or other packages that they inherit classes and functions from.**
* **Often, packages require certain versions from their dependencies. So… How do you figure out which versions of the packages to install so that all the packages agree with each other? Use version control software (anaconda/conda) to maintain your own environment. I will provide an environment for this class (culabenv2023clean.yml).**

Where will I run iPython notebooks with python?
Lots of options!! Plan to set-up and test code THIS WEEK.

- Install on your own computer. I recommend using Anaconda. Instructions for replicating my installation available on Google Drive (Mac Only).
- Run in the cloud using Google Colab

https://colab.research.google.com/notebooks/welcome.ipynb

- Use Jupyter Hub on a supercomputer.

1) NCAR/NSF Systems. All graduate students can apply for accounts. https://www2.cisl.ucar.edu/user-support/allocations/university-allocations/university-gradpostdoc-request

2) CU. Apply for an account with Research Computing here: https://rcamp.rc.colorado.edu/accounts/account-request/create/organization

Lecture, Application Lab, and Homework Schedule

*Available on Google Drive

*Subject to change

	Tuesday	Thursday
January	**January 17** Complete pre-class survey, Set up python environment, Read syllabus.	**January 19** 1. Introductions/Basic statistics/Bayes Theorem (Barnes 1.1-1.2)
	January 24 2. Statistical Significance Testing /Hypothesis testing/Resampling/Monte Carlo (Barnes 1.3-1.5) HW#1 assigned	**January 26** Application LAB #1 Basic Statistics and Hypothesis testing
	January 31 Applications LAB #1 cont.	**February 2** 3. Compositing/Other distributions/Non-parametric tests (Barnes 1.6-1.8) HW#1 due HW#2 assigned
February	**February 7** 4. Regression (Barnes 2.1-2.2)	**February 9** 5. Autocorrelation/Autoregressive model/Sample Size/Multiple Regression (Barnes 2.3-2.4)
	February 14 Applications LAB #2 Regression/AR1	**February 16** Applications LAB #2 cont.
	February 21 6. EOFs via Eigenanalysis/SVD (Barnes 3.1.1-3.1.4) HW#2 due, HW#3 assigned	**February 23** 7. EOFs with actual data (Barnes 3.1.5)
	February 28 8. More EOFs with Dr. Nicola Maher	**March 2** Applications LAB #3 – Matrix methods/EOFs
March	**March 7** Applications LAB #3 cont.	**March 9** 9. Harmonic analysis; power spectra (Barnes 4.1.1-4.1.2) HW#3 due; HW#4 assigned
	March 14 10. Fourier Transforms/Significance testing of spectral peaks/Data windows (Barnes 4.1.3-4.1.5)	**March 16** Applications LAB #4 – Timeseries analysis/Power spectra
	March 21 Applications LAB #4 cont.	**March 23** PRESENTATIONS on Homework #2, #3 HW#4 due
	SPRING BREAK – NO CLASS	
April/May	**April 4** 11. Filtering (Barnes 4.1.6; Hartmann 7), HW#5 assigned	**April 6** 12. Finish Filtering (Barnes 4.1.6; Hartmann 7)
	April 11 Applications LAB #5 – Timeseries analysis/Filtering	**April 13** Applications LAB #5 cont. HW#5 due, HW#6 assigned
	April 18 13. Machine Learning Overview	**April 20** 14. Machine Learning – SOMs, clustering
	April 25 Applications LAB #6: Machine Learning	**April 27** Applications LAB #6 cont.
	May 2 15. More machine learning with Dr. Nicola Bodini Homework #6 due	**May 4** PRESENTATIONS: Homework #4, #5, #6

Spring 2023: ATOC5860 Objective Data Analysis

Classes in yellow are entirely "learning by doing" application labs in small groups

Last updated: January 13, 2022

TWO GUEST LECTURES BY NICOLAS ☺
Dr. Nicola Maher (left)
Dr. Nicola Bodini (right)

https://nicolamaher.weebly.com/

https://www.nrel.gov/research/staff/nicola-bodini.html

Class Protocols:

1) Arrive 5 minutes early. I'll always be there 15 minutes early.
2) Ask questions, communicate!
3) Be Respectful, Be Engaged, and Be Prepared.

ANY QUESTIONS???
IDEAS ON HOW TO MAKE THIS CLASS WORK WELL??

Who am I?

Associate Professor Jen Kay,
ATOC and CIRES
Please call me "Jen"

CU Faculty 2014-present
NCAR Research Scientist 2009-2013
NCAR/CSU Postdoc 2007-2009
Ph.D. 2006 from the University of Washington

Research Interests: Climate Change and Variability,
Clouds, Large-scale circulation, Ice
Author of 100 Peer-reviewed Publications.
Reviewer of 100s of publications, Editor Journal of Climate
Highly Cited Researcher (2021)

Non-science: Doggies, Hiking, Skiing, Bread, Gardening

Who are you?

Name
Department
Research Interests
Goals for this Class
Something non-science

What are statistics good for?

- Statistics provide evidence to support (or not support) a hypothesis/physical mechanism.
- Statistics are an objective reality check for the scientist (YOU!).
- Statistics help you find patterns in the noise: "big data" is the new buzz word… but what do you do with all of that data??

What are statistics NOT good for?

- **Replacing good scientific thinking:** Statistics cannot "prove" anything. All statistics have a chance of "getting it wrong".
- **Being blindly applied:** Every analysis endeavor is slightly different and has its own nuances. Think critically. Don't be a statistics/big data zombie.
- **Making-up for a lack of data:** Statistical analysis will not replace having enough good data. If you don't have enough data to conclude anything, you don't have enough data to conclude anything. Admit defeat and get some more data!!

A Few Rules of Data Analysis

- **Look at Your Data:** *always* look at your data before, during, and after your analysis. Look at the values, plot the values, make sure they make sense. You will hear me say this over and over again this class….

- **Document what you have done and why; make sure what you are doing is reproducible:** commenting your code will slow you down now but save you time later and make your work easier to share.

- **Pay Attention To What The Numbers Say (Or Don't Say):** Even good scientists fall into the trap of ignoring statistics that contradict their beliefs. Take care that you don't fool yourself into seeing what you want to see.

- **Be Careful & Be Skeptical:** There are countless examples of people abusing statistics or just not knowing any better. Be aware of how this can be done, keep an eye out for it, and speak up when you see it. As a reviewer of others' work – you will need to be able to critically evaluate the application of the methods we discuss in this class.

Basic Statistics – Mean, Variance

- Mean, Variance. Contrast sample and population. Notation in this class follows Barnes: μ= population mean; σ=population standard deviation, χ=sample mean; S = sample standard deviation

The sample mean is defined as

$$\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad (5)$$

where the bar denotes the time mean and the subscript i denotes the time step (we will be working mainly with time series here).

The sample mean $\overline{x}$ is an unbiased estimate of the true mean μ.

In other words: if you draw an infinite number of *samples* from the same time series, then the actual population mean of all of the sample means ($\mu_{\overline{x}}$) is equal to the population mean (μ).

The sample variance is defined as:

$$\overline{x'^2} = \frac{1}{N-1}\sum_{i=1}^{N} (x_i - \overline{x})^2 \qquad (6)$$

where the prime denotes departures from the mean.

The sample standard deviation is defined as:

$$s = \sqrt{\overline{x'^2}} \qquad (7)$$

Contrast Frequentist vs. Bayesian Approaches

There are two general philosophies of thought on statistics: (a) frequentist approach, and (b) the Bayesian approach.

Frequentist: if you give an event many opportunities to occur, the probability of occurrence is the

$$\frac{\#\text{ of occurrences}}{\#\text{ of opportunities}} \tag{62}$$

This approach works well when you can repeat an experiment many, many times.

Bayesian: this approach is named after the frequent use of Bayes theorem, which takes into account a priori information that may not be useable by a frequentist.

Bayes Theorem Example: Testing Accuracy During a Pandemic

Background rates of COVID are 90% negative, 10% positive

COVID tests are accurate 80% of the time, but fail 20% of the time.

You go and get a COVID test. You test negative.

What is the probability that you are actually negative?

Many would answer is 80%, but this answer does not take into account background rates of COVID.

Bayesian approaches take into account the background rate of infection, information that frequentist approaches cannot use.

Bayes Theorem – Definition

Bayes' Theorem takes $\boldsymbol{Pr}(A|B)$ and turns it into $\boldsymbol{Pr}(B|A)$.

Let $E_i, i = 1, 2, 3...N$ be a set of N events such that the set E_i includes all possible possibilities in a set S and the events are mutually exclusive. Then, for any event B, with $\boldsymbol{Pr}(B) > 0$

$$\boldsymbol{Pr}(E_j|B) = \frac{\boldsymbol{Pr}(B|E_j)\,\boldsymbol{Pr}(E_j)}{\sum_{i=1}^{N} \boldsymbol{Pr}(B|E_i)\,\boldsymbol{Pr}(E_i)} \tag{23}$$

https://www.youtube.com/watch?v=HZGCoVF3YvM

Bayes Theorem Example, Testing accuracy during pandemic

1. Define Variables and State What You Know		
Background Information		
P(N)	Percent of Population Negative	90%
P(P)	Percent of Population Positive	10%
Reliability of test		
P(T|N)	Probability that test accurate (you test negative, you are negative)	80%
P(T|P)	Probability that test fails (you test negative, you are positive)	20%
2. State what we want to know		
What is the probability of being negative given that the test told you negative?		P(N|T)
P (N|T) is	P(T/N)*P(N) / (P(T/N)*P(N)+P(T/P)*P(P))	
	top = test correct	
	bottom = all possible outcomes	
P(N|T)	97%	

Bayes Theorem Example, Testing accuracy during pandemic

1. Define Variables and State What You Know		
Background Information		
P(N)	Percent of Population Negative	1%
P(P)	Percent of Population Positive	99%
Reliability of test		
P(T|N)	Probability that test accurate (you test negative, you are negative)	80%
P(T|P)	Probability that test fails (you test negative, you are positive)	20%
2. State what we want to know		
What is the probability of being negative given that the test told you negative?		P(N|T)
P (N|T) is	P(T/N)*P(N) / (P(T/N)*P(N)+P(T/P)*P(P))	
	top = test correct	
	bottom = all possible outcomes	
P(N|T)	4%	

Bayes Theorem Example, Testing accuracy during pandemic

1. Define Variables and State What You Know		
Background Information		
P(N)	Percent of Population Negative	50%
P(P)	Percent of Population Positive	50%
Reliability of test		
P(T|N)	Probability that test accurate (you test negative, you are negative)	80%
P(T|P)	Probability that test fails (you test negative, you are positive)	20%
2. State what we want to know		
What is the probability of being negative given that the test told you negative?		P(N|T)
P (N|T) is	P(T/N)*P(N) / (P(T/N)*P(N)+P(T/P)*P(P))	
	top = test correct	
	bottom = all possible outcomes	
P(N|T)	80%	

Bayes Theorem – Another Example (Barnes 1.2.2.2)
REVIEW ON YOUR OWN

A cab was involved in a hit and run accident at night. Two cab companies, the Green and the Blue, operate in the city. You are given the following data:

- 85% of the cabs in the city are Green and 15% are Blue.
- A witness identified the cab as Blue. The court tested the reliability of the witness under the circumstances that existed the night of the accident and concluded that the witness correctly identified each one of the two colors 80% of the time and failed 20% of the time.

What is the probability that the cab involved in the accident was Blue rather than Green?

Your instincts might tell you 80% - but then you wouldn't be taking into account the background rate of the cabs in the city.

Applying Bayes Theorem: we get 41% probability the cab was blue.

Note: If you taken a frequentist approach (not taking into account the witness), your answer would have been 15%…

Introducing the Normal Distribution

- Probability Density/Distribution Function (PDF) vs. Cumulative Density/Distribution Function (CDF)

The probability density function for a variable x that is normally distributed about its mean is given by:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)} \tag{68}$$

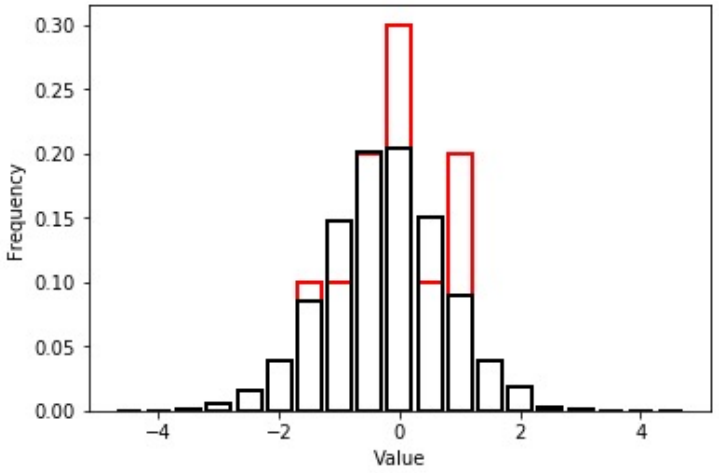
The associated cumulative distribution function is given as:

$$F(b) = \int_{-\infty}^{b} \frac{1}{\sigma\sqrt{2\pi}} e^{\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)} dx \tag{69}$$

Impact of sample size (N) on the normal distribution. What does a normal distribution look like with 10 randomly selected values (N=10, red) vs. 3000 randomly selected values (N=3000, black)?

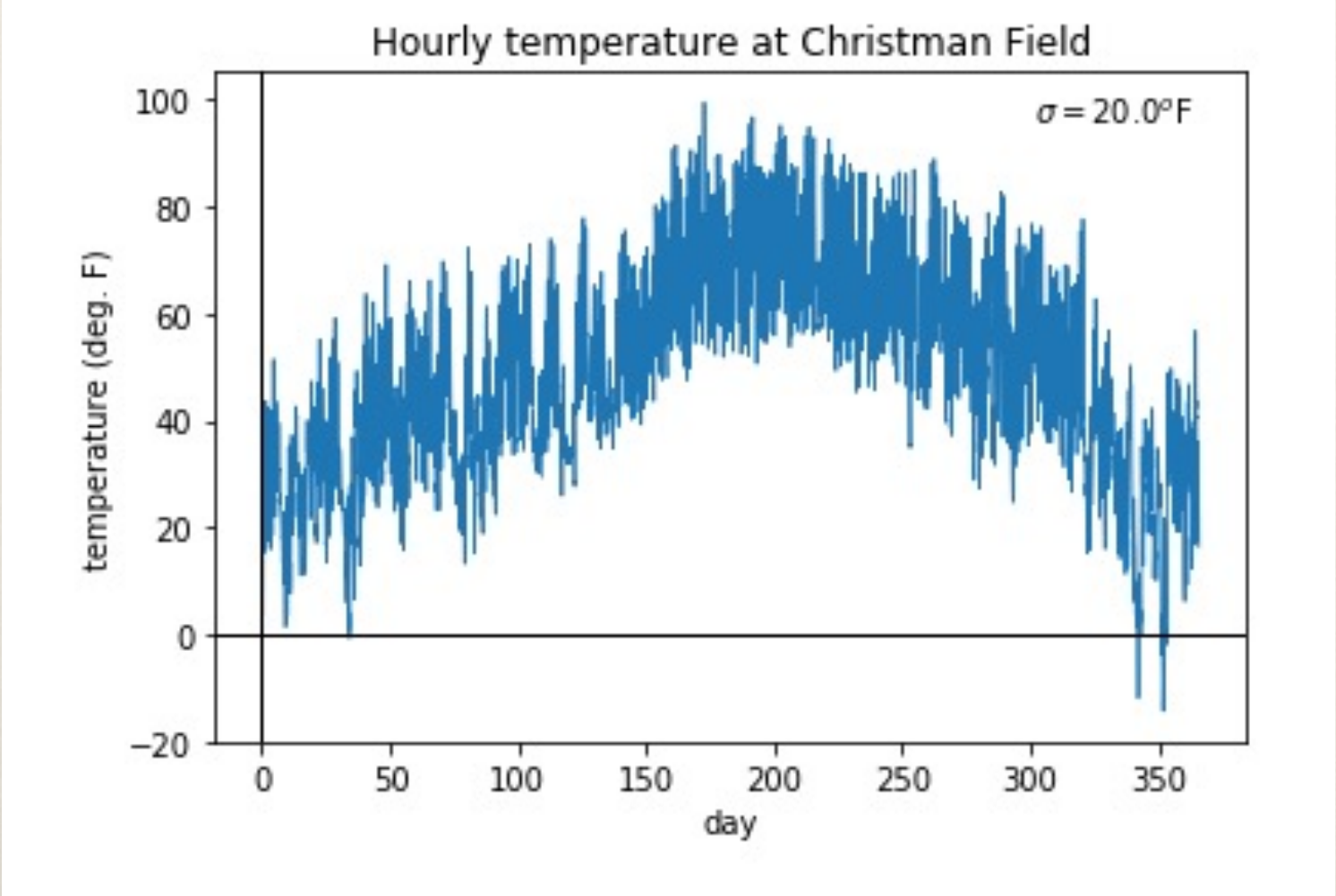
- Run ipython notebook – what_is_pdf.ipynb

In [5]:
```
## Plot Normalized Probability Distribution Function
hx = np.histogram(x,xinc)
hy = np.histogram(y,xinc)
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.bar(hx[1][:-1],hx[0].astype(float)/np.size(x),edgecolor = 'r', color = [], width = .4, linewidth = 2)
plt.bar(hy[1][:-1],hy[0].astype(float)/np.size(y),edgecolor = 'k', color = [], width = .4, linewidth = 2)
plt.show()
```



Variance Example… using Temperature data from Christman Field, Colorado (near Fort Collins).

- Run ipython notebook – variance_example.ipynb



A notebook to look at, especially if newer to python

- Run ipython notebook – python_gotchas.ipynb

ATOC5860 - some python gotchas - Lecture #1

Prof. Kay (CU)

Last updated: January 13, 2023

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd  # library for data analysis for text files (everything but netcdf files)
%matplotlib inline
```

In [2]:
```
## python gotcha #1 -- indexing
foo=np.linspace(0,10,11)
print(foo)
print(foo.shape)

## python is 0 based
print(foo[0])

## indexing over the first 5 values of an array only selects the first 4 values of an array
print(foo[0:4])
```

```
[ 0.  1.  2.  3.  4.  5.  6.  7.  8.  9. 10.]
(11,)
0.0
[0. 1. 2. 3.]
```

In [3]:
```
## python gotcha #2 -- copying vs. linking arrays

## if you want to copy an array
foo_new = foo.copy()

## if you want to link an array
foo_new_linked = foo

foo[0]=10
print(foo)
print(foo_new)
print(foo_new_linked)
```

```
[10.  1.  2.  3.  4.  5.  6.  7.  8.  9. 10.]
[ 0.  1.  2.  3.  4.  5.  6.  7.  8.  9. 10.]
[10.  1.  2.  3.  4.  5.  6.  7.  8.  9. 10.]
```

In [4]:
```
## python slick combining text and numbers (Thanks Devon Dunmire (CU))

print(f'this is {foo}')
print('this is {foo}')
```

```
this is [10.  1.  2.  3.  4.  5.  6.  7.  8.  9. 10.]
this is {foo}
```

Do you have a python gotcha to add? Send it to Professor Kay :)

"TO DO" **This Week**

- Complete pre-class survey - https://forms.gle/MGh8b3JK6oiwVBSJ9
- Read Barnes 1.1 – 1.2
- Install Python (if needed), Build the environment for this class, and Test running the three Jupyter Notebooks available in the class Google Drive in the folder called "lecture_code/lecture1": variance_example.ipynb; python_gotchas.ipynb; whatispdf.ipynb
- The python environment I use and information for installing it on a Mac computer is available in a folder called "python_installation_computing_help" on Google Drive. Google is your friend, as are your fellow students, and (last resort) me.