

# Introduction

Francisco Villamil

Applied Quantitative Methods II

IC3JM, Spring 2026

# Course overview

- This is the second part of the quantitative methods sequence

# Course overview

- This is the second part of the quantitative methods sequence
- Focus on **applying** statistical tools in practice

# Course overview

- This is the second part of the quantitative methods sequence
- Focus on **applying** statistical tools in practice
- Less theory, more hands-on work with data

# Course overview

- This is the second part of the quantitative methods sequence
- Focus on **applying** statistical tools in practice
- Less theory, more hands-on work with data
- Goal: go from research question to answer

# What will you learn?

- How to choose the right model for your question

# What will you learn?

- How to choose the right model for your question
- How to interpret and visualize model results

# What will you learn?

- How to choose the right model for your question
- How to interpret and visualize model results
- How to evaluate whether a model is appropriate



# What will you learn?

- How to choose the right model for your question
- How to interpret and visualize model results
- How to evaluate whether a model is appropriate
- How to work with different types of data (panel, spatial, etc.)

# What will you learn?

- How to choose the right model for your question
- How to interpret and visualize model results
- How to evaluate whether a model is appropriate
- How to work with different types of data (panel, spatial, etc.)
- Best practices in computing and reproducibility

# Course structure

<b>Feb 5</b>	Introduction
<b>Feb 12-19</b>	i2i
<b>Feb 26</b>	i3i
<b>Mar 5</b>	i4i
<b>Mar 12-19</b>	i5i
<b>Mar 26 &amp; Apr 9</b>	i6i
<b>Apr 16</b>	i7i
<b>Apr 23</b>	Project presentations
<b>Apr 30</b>	Advanced topics

# Course structure

<b>Feb 5</b>	Introduction
<b>Feb 12-19</b>	i2i
<b>Feb 26</b>	i3i
<b>Mar 5</b>	i4i
<b>Mar 12-19</b>	i5i
<b>Mar 26 &amp; Apr 9</b>	i6i
<b>Apr 16</b>	i7i
<b>Apr 23</b>	Project presentations
<b>Apr 30</b>	Advanced topics

# Course structure

<b>Feb 5</b>	Introduction
<b>Feb 12-19</b>	i2i
<b>Feb 26</b>	i3i
<b>Mar 5</b>	i4i
<b>Mar 12-19</b>	i5i
<b>Mar 26 &amp; Apr 9</b>	i6i
<b>Apr 16</b>	i7i
<b>Apr 23</b>	Project presentations
<b>Apr 30</b>	Advanced topics

# Course structure

<b>Feb 5</b>	Introduction
<b>Feb 12-19</b>	i2i
<b>Feb 26</b>	i3i
<b>Mar 5</b>	i4i
<b>Mar 12-19</b>	i5i
<b>Mar 26 &amp; Apr 9</b>	i6i
<b>Apr 16</b>	i7i
<b>Apr 23</b>	Project presentations
<b>Apr 30</b>	Advanced topics

# Course structure

<b>Feb 5</b>	Introduction
<b>Feb 12-19</b>	i2i
<b>Feb 26</b>	i3i
<b>Mar 5</b>	i4i
<b>Mar 12-19</b>	i5i
<b>Mar 26 &amp; Apr 9</b>	i6i
<b>Apr 16</b>	i7i
<b>Apr 23</b>	Project presentations
<b>Apr 30</b>	Advanced topics

# Course structure

<b>Feb 5</b>	Introduction
<b>Feb 12-19</b>	i2i
<b>Feb 26</b>	i3i
<b>Mar 5</b>	i4i
<b>Mar 12-19</b>	i5i
<b>Mar 26 &amp; Apr 9</b>	i6i
<b>Apr 16</b>	i7i
<b>Apr 23</b>	Project presentations
<b>Apr 30</b>	Advanced topics



# Course structure

<b>Feb 5</b>	Introduction
<b>Feb 12-19</b>	i2i
<b>Feb 26</b>	i3i
<b>Mar 5</b>	i4i
<b>Mar 12-19</b>	i5i
<b>Mar 26 &amp; Apr 9</b>	i6i
<b>Apr 16</b>	i7i
<b>Apr 23</b>	Project presentations
<b>Apr 30</b>	Advanced topics

# Evaluation

- Problem sets (20%)
  - Started in class, finished at home
  - Short deadlines
- Proposal presentation and peer review (10% + 10%)
- Final essay (30%)
  - Small research note (max 3,000 words)
  - Original data analysis using R
- Exam (30%)

# The Big Picture

# The research process

**Theory  $\longleftrightarrow$  Data Generating Process  $\longleftrightarrow$  Data**

- Theories make claims about how the world works

# The research process

**Theory  $\longleftrightarrow$  Data Generating Process  $\longleftrightarrow$  Data**

- Theories make claims about how the world works
- These claims imply certain patterns in data

# The research process

**Theory  $\longleftrightarrow$  Data Generating Process  $\longleftrightarrow$  Data**

- Theories make claims about how the world works
- These claims imply certain patterns in data
- We observe data and try to learn about the underlying process

# What is a Data Generating Process (DGP)?

- The rules that govern how data comes to exist

# What is a Data Generating Process (DGP)?

- The rules that govern how data comes to exist
- Includes:



# What is a Data Generating Process (DGP)?

- The rules that govern how data comes to exist
- Includes:
  - The social or political process we study

# What is a Data Generating Process (DGP)?

- The rules that govern how data comes to exist
- Includes:
  - The social or political process we study
  - How observations end up in our dataset

# What is a Data Generating Process (DGP)?

- The rules that govern how data comes to exist
- Includes:
  - The social or political process we study
  - How observations end up in our dataset
- We never observe the DGP directly

# What is a Data Generating Process (DGP)?

- The rules that govern how data comes to exist
- Includes:
  - The social or political process we study
  - How observations end up in our dataset
- We never observe the DGP directly
- We use statistical models to make inferences about it

# Why do we need statistics?

- Our theories deal with processes, not just data

# Why do we need statistics?

- Our theories deal with processes, not just data
- Data is a window into the underlying process

# Why do we need statistics?

- Our theories deal with processes, not just data
- Data is a window into the underlying process
- Statistics helps us:

# Why do we need statistics?

- Our theories deal with processes, not just data
- Data is a window into the underlying process
- Statistics helps us:
  - Separate signal from noise



# Why do we need statistics?

- Our theories deal with processes, not just data
- Data is a window into the underlying process
- Statistics helps us:
  - Separate signal from noise
  - Quantify uncertainty

# Why do we need statistics?

- Our theories deal with processes, not just data
- Data is a window into the underlying process
- Statistics helps us:
  - Separate signal from noise
  - Quantify uncertainty
  - Make valid inferences

# Sources of uncertainty

- **Sampling uncertainty:** We observe a sample, not the population

# Sources of uncertainty

- **Sampling uncertainty:** We observe a sample, not the population
- **Theoretical uncertainty:** Our theories are simplifications

# Sources of uncertainty

- **Sampling uncertainty:** We observe a sample, not the population
- **Theoretical uncertainty:** Our theories are simplifications
- **Fundamental uncertainty:** Some processes are inherently random

# Sources of uncertainty

- **Sampling uncertainty:** We observe a sample, not the population
- **Theoretical uncertainty:** Our theories are simplifications
- **Fundamental uncertainty:** Some processes are inherently random

# Sources of uncertainty

- **Sampling uncertainty:** We observe a sample, not the population
- **Theoretical uncertainty:** Our theories are simplifications
- **Fundamental uncertainty:** Some processes are inherently random
- All of these create “noise” in our data

# Sources of uncertainty

- **Sampling uncertainty:** We observe a sample, not the population
- **Theoretical uncertainty:** Our theories are simplifications
- **Fundamental uncertainty:** Some processes are inherently random
  
- All of these create “noise” in our data
- Statistical models help us deal with this noise



# The logic of statistical inference

- **Probability theory:** Given a known process, what data will we see?

# The logic of statistical inference

- **Probability theory:** Given a known process, what data will we see?

# The logic of statistical inference

- **Probability theory:** Given a known process, what data will we see?
- **Statistical inference:** Given observed data, what can we learn about the process?

# The logic of statistical inference

- **Probability theory:** Given a known process, what data will we see?
- **Statistical inference:** Given observed data, what can we learn about the process?

# The logic of statistical inference

- **Probability theory:** Given a known process, what data will we see?
- **Statistical inference:** Given observed data, what can we learn about the process?
- We're doing the reverse: from data back to process

# Version Control and Git

# The problem: managing files over time

- Have you ever had files like this?

# The problem: managing files over time

- Have you ever had files like this?

→ `thesis_v1.docx`



# The problem: managing files over time

- Have you ever had files like this?
  - `thesis_v1.docx`
  - `thesis_v2_final.docx`

# The problem: managing files over time

- Have you ever had files like this?
  - `thesis_v1.docx`
  - `thesis_v2_final.docx`
  - `thesis_v2_final_REAL.docx`

# The problem: managing files over time

- Have you ever had files like this?
  - `thesis_v1.docx`
  - `thesis_v2_final.docx`
  - `thesis_v2_final_REAL.docx`
  - `thesis_v2_final_REAL_submitted.docx`

# The problem: managing files over time

- Have you ever had files like this?
  - `thesis_v1.docx`
  - `thesis_v2_final.docx`
  - `thesis_v2_final_REAL.docx`
  - `thesis_v2_final_REAL_submitted.docx`

# The problem: managing files over time

- Have you ever had files like this?
  - `thesis_v1.docx`
  - `thesis_v2_final.docx`
  - `thesis_v2_final_REAL.docx`
  - `thesis_v2_final_REAL_submitted.docx`
- What changed between versions?

# The problem: managing files over time

- Have you ever had files like this?
  - `thesis_v1.docx`
  - `thesis_v2_final.docx`
  - `thesis_v2_final_REAL.docx`
  - `thesis_v2_final_REAL_submitted.docx`
- What changed between versions?
- Which version has the correct analysis?

# The problem: managing files over time

- Have you ever had files like this?
  - `thesis_v1.docx`
  - `thesis_v2_final.docx`
  - `thesis_v2_final_REAL.docx`
  - `thesis_v2_final_REAL_submitted.docx`
- What changed between versions?
- Which version has the correct analysis?
- How do you collaborate without overwriting each other's work?

# Version control: a better way

**Version control** is a system that records changes to files over time

- One file, complete history



# Version control: a better way

**Version control** is a system that records changes to files over time

- One file, complete history
- Every change is recorded with a description

# Version control: a better way

**Version control** is a system that records changes to files over time

- One file, complete history
- Every change is recorded with a description
- Can go back to any previous state

# Version control: a better way

**Version control** is a system that records changes to files over time

- One file, complete history
- Every change is recorded with a description
- Can go back to any previous state
- Multiple people can work simultaneously

# Why version control for research?

- **Reproducibility:** Track exactly what you did and when

# Why version control for research?

- **Reproducibility:** Track exactly what you did and when
- **Backup:** Your work is safely stored, even if your laptop dies

# Why version control for research?

- **Reproducibility:** Track exactly what you did and when
- **Backup:** Your work is safely stored, even if your laptop dies
- **Collaboration:** Work with others without email chains of files

# Why version control for research?

- **Reproducibility:** Track exactly what you did and when
- **Backup:** Your work is safely stored, even if your laptop dies
- **Collaboration:** Work with others without email chains of files
- **Transparency:** Share your code with the research community

# Why version control for research?

- **Reproducibility:** Track exactly what you did and when
- **Backup:** Your work is safely stored, even if your laptop dies
- **Collaboration:** Work with others without email chains of files
- **Transparency:** Share your code with the research community



# Why version control for research?

- **Reproducibility:** Track exactly what you did and when
- **Backup:** Your work is safely stored, even if your laptop dies
- **Collaboration:** Work with others without email chains of files
- **Transparency:** Share your code with the research community
- Many journals now require or encourage sharing code via GitHub

# Git and GitHub

## **Git**

- A version control system
- Runs locally on your computer
- Tracks changes to files

## **GitHub**

- A web platform that hosts Git repositories
- Stores your code online
- Enables sharing and collaboration

# The basic Git workflow

1. **Make changes** to your files (write code, edit text)

# The basic Git workflow

1. **Make changes** to your files (write code, edit text)
2. **Stage** the changes you want to save

# The basic Git workflow

1. **Make changes** to your files (write code, edit text)
2. **Stage** the changes you want to save
  - “These are the files I want to include in my next snapshot”

# The basic Git workflow

1. **Make changes** to your files (write code, edit text)
2. **Stage** the changes you want to save
  - “These are the files I want to include in my next snapshot”
3. **Commit** the staged changes with a message

# The basic Git workflow

1. **Make changes** to your files (write code, edit text)
2. **Stage** the changes you want to save
  - “These are the files I want to include in my next snapshot”
3. **Commit** the staged changes with a message
  - A snapshot of your project at this moment

# The basic Git workflow

1. **Make changes** to your files (write code, edit text)
2. **Stage** the changes you want to save
  - “These are the files I want to include in my next snapshot”
3. **Commit** the staged changes with a message
  - A snapshot of your project at this moment
4. **Push** your commits to GitHub



# The basic Git workflow

1. **Make changes** to your files (write code, edit text)
2. **Stage** the changes you want to save
  - “These are the files I want to include in my next snapshot”
3. **Commit** the staged changes with a message
  - A snapshot of your project at this moment
4. **Push** your commits to GitHub
  - Upload your local changes to the cloud

# Ways to use Git

- **GitHub web interface:** Create repos, upload files, edit directly

# Ways to use Git

- **GitHub web interface:** Create repos, upload files, edit directly
  - Simple but limited

# Ways to use Git

- **GitHub web interface:** Create repos, upload files, edit directly
  - Simple but limited
- **Command line:** Most powerful and flexible

# Ways to use Git

- **GitHub web interface:** Create repos, upload files, edit directly
  - Simple but limited
- **Command line:** Most powerful and flexible
  - `git add`, `git commit`, `git push`

# Ways to use Git

- **GitHub web interface:** Create repos, upload files, edit directly
  - Simple but limited
- **Command line:** Most powerful and flexible
  - `git add`, `git commit`, `git push`
- **RStudio:** Built-in Git integration

# Ways to use Git

- **GitHub web interface:** Create repos, upload files, edit directly
  - Simple but limited
- **Command line:** Most powerful and flexible
  - `git add`, `git commit`, `git push`
- **RStudio:** Built-in Git integration
  - Point-and-click interface

# Ways to use Git

- **GitHub web interface:** Create repos, upload files, edit directly
  - Simple but limited
- **Command line:** Most powerful and flexible
  - `git add`, `git commit`, `git push`
- **RStudio:** Built-in Git integration
  - Point-and-click interface



# Ways to use Git

- **GitHub web interface:** Create repos, upload files, edit directly
  - Simple but limited
- **Command line:** Most powerful and flexible
  - `git add`, `git commit`, `git push`
- **RStudio:** Built-in Git integration
  - Point-and-click interface
- All do the same thing—choose what works for you

# Problem Set 1

- Create a GitHub account (if you don't have one)

# Problem Set 1

- Create a GitHub account (if you don't have one)
- Create a **public** repository for this course

# Problem Set 1

- Create a GitHub account (if you don't have one)
- Create a **public** repository for this course
- Set up your README and folder structure

# Problem Set 1

- Create a GitHub account (if you don't have one)
- Create a **public** repository for this course
- Set up your README and folder structure
- Create a simple .R file

# Problem Set 1

- Create a GitHub account (if you don't have one)
- Create a **public** repository for this course
- Set up your README and folder structure
- Create a simple .R file

# Problem Set 1

- Create a GitHub account (if you don't have one)
- Create a **public** repository for this course
- Set up your README and folder structure
- Create a simple .R file
  
- This repository is where you'll submit all your assignments

# Problem Set 1

- Create a GitHub account (if you don't have one)
  - Create a **public** repository for this course
  - Set up your README and folder structure
  - Create a simple .R file
- 
- This repository is where you'll submit all your assignments
  - Detailed instructions in the problem set document



# What makes a good analysis?

- Clear research question

# What makes a good analysis?

- Clear research question
- Appropriate data for the question

# What makes a good analysis?

- Clear research question
- Appropriate data for the question
- Right statistical model for the data

# What makes a good analysis?

- Clear research question
- Appropriate data for the question
- Right statistical model for the data
- Correct interpretation of results

# What makes a good analysis?

- Clear research question
- Appropriate data for the question
- Right statistical model for the data
- Correct interpretation of results
- Honest about limitations and uncertainty

# Looking ahead

- Next session: Applied regression in depth

# Looking ahead

- Next session: Applied regression in depth
- How to set up a regression analysis

# Looking ahead

- Next session: Applied regression in depth
- How to set up a regression analysis
- How to interpret coefficients correctly



# Looking ahead

- Next session: Applied regression in depth
- How to set up a regression analysis
- How to interpret coefficients correctly
- Common pitfalls and how to avoid them

## For next week

- Read Urdinez & Cruz (2020), chapters 1-5
- Review your notes on OLS from AQMSS-I
- Start Problem Set 1
  
- Check Aula Global for additional materials

Questions?