

ÖZYEĞİN ÜNİVERSİTESİ

DS 530
Fairness and Interpretability in Data
Science

ENİS KAYIŞ

Instructor

- ▶ Enis Kayış
 - ▶ Assistant Professor in Industrial Engineering
 - ▶ Email: enis.kayis@ozyegin.edu.tr
 - ▶ Phone: +90 (216) 594-9370
 - ▶ Office: 215
 - ▶ Office Hours: By Appointment
- ▶ Short Bio:
 - ▶ Education:
 - ▶ PhD in Management Science and Engineering, Stanford University, 2009
 - Essays on Procurement Contracting
 - ▶ MS in Statistics, Stanford University, 2007
 - ▶ BS in Industrial Engineering and Mathematics, Bogazici University, 2002
 - ▶ Work Experience:
 - ▶ Research Scientist at HP Labs, since 2009
 - Research projects in areas including demand estimation, product portfolio management and pricing, healthcare operations, and forecasting

Course Description

- ▶ With increasing complexity of data science methods, fairness and interpretability of the employed methods have to be treated as a central concern before implementation. In this course, how to identify potential sources of bias in different contexts as well as quantification of bias with alternative fairness metrics are discussed using real-life applications. The notion of model interpretability and different classes of interpretable models are explained. State-of-the art techniques are introduced to increase fairness and interpretability of the traditional machine learning methods.
- ▶ **Prerequisites:** Introductory knowledge in data science and ability to write computer programs.

Suggested Readings

- ▶ **Suggested Online Textbooks:**
 - ▶ Fairness and Machine Learning Book:
 - <https://fairmlbook.org/>
 - ▶ Interpretable Machine Learning:
 - <https://christophm.github.io/interpretable-ml-book/>
- ▶ Additional material posted to the Course Website (LMS)

Grading Policy

- ▶ **Research Paper Presentations: 15%**
 - ▶ An oral presentation in the classroom on the main research findings of a research paper in this field.
- ▶ **Case Discussions: 5%**
 - ▶ A short oral presentation in the classroom about a documented case that require fairness and/or interpretability considerations to be incorporated.
- ▶ **Midterm Exam I: 30%**
 - ▶ Date: 25 November 2023
- ▶ **Midterm Exam II: 30%**
 - ▶ Date: 16 December 2023
- ▶ **Final Project: 20%**
 - ▶ More on this later

Case and Paper Discussions

▶ Case Discussion:

- ▶ Individual Exercise
- ▶ Time: 5 minutes
- ▶ The case for interpretability or fairness in data science using popular news articles, research papers, etc.
- ▶ One paragraph summary due before class (Max 1000 characters)

▶ Research Paper Discussion:

- ▶ Team (2 people) or Individual Exercise
- ▶ Time: 15 minutes
- ▶ The case for interpretability or fairness in data science using popular news articles, research papers, etc.
- ▶ One paragraph summary due before class (Max 1000 characters)

Final Project

- ▶ A written report for the solution of a fairness or interpretability concern is due by the end of semester.
- ▶ Students can work in groups of at most three.
- ▶ Send submissions by the deadline via LMS link.

Course Material (subject to change)

WEEKLY SUBJECTS (TENTATIVE)		
Week	Date	Subject
1	02-06.10.2023	Introduction to Fairness and Interpretability
2	09-13.10.2023	Notions for Interpretable Models
3	16-20.10.2023	Interpretable Data Science Models I
4	23-27.10.2023	Interpretable Data Science Models II
5	30.10-03.11.2023	Global Model-Agnostic Methods I
6	06-10.11.2023	Global Model-Agnostic Methods II
7	13-17.11.2023	Local Model-Agnostic Methods I
8	20-24.11.2023	Local Model-Agnostic Methods II
9	27.11-01.12.2023 <i>(Withdrawal Week)</i>	Midterm Exam I
10	04-08.12.2023	Introduction to Fairness
11	11-15.12.2023	Causes of Bias and Fairness Metrics
12	18-22.12.2023	Debiasing Data
13	25-29.12.2023	Algorithms to achieve fairness
14	01-05.01.2024	Midterm Exam II

Caution

- ▶ How is this course different than typical DS classes you've taken before?
- ▶ I will assume all the participants knows:
- ▶ Classical methods/algorithms/models (e.g., regression, decision trees, SVM, NN, etc.)
- ▶ Assumptions behind these methods
- ▶ How to implement these methods

Artificial Intelligence

- ▶ OECD's five principles on AI:
 1. AI should benefit people and the planet by driving inclusive growth, sustainable development and well-being.
 2. AI systems should be designed in a way that **respects the rule of law, human rights, democratic values and diversity**, and they should include appropriate safeguards – for example, enabling human intervention where necessary – **to ensure a fair and just society**.
 3. There should be **transparency and responsible disclosure** around AI systems to ensure that people understand AI-based outcomes and can challenge them.
 4. AI systems must function in a robust, secure and safe way throughout their life cycles and potential risks should be continually assessed and managed.
 5. Organizations and individuals developing, deploying or operating AI systems should be held accountable for their proper functioning in line with the above principles.

Responsible AI (RAI)

CEOs should provide guidance to help analytics teams build and use AI responsibly.

Translate company values into AI development and dig deep by asking analytics teams questions in key areas.
<ul style="list-style-type: none">Clarify how values translate into the selection of AI applications, such as what processes to automate.	Data acquisition <i>Are we aligned with our stakeholders' expectations for the use of their data?</i>
<ul style="list-style-type: none">Provide guidance on definitions and metrics for evaluating AI for bias and fairness.	Data-set suitability <i>Do data sets reflect real-world populations? Have they included data that are relevant to minority groups?</i>
<ul style="list-style-type: none">Advise on the hierarchy of company values and role of diversity in talent selection.	AI-output fairness <i>Is fairness considered at every point in the development process, including data selection, feature selection, and model building and monitoring?</i>
	Regulatory compliance and engagement <i>Do we have compliance built into our workflows, and do we share our market and technical acumen in the development of new regulations?</i>
	AI-model explainability <i>Are we using the simplest performance model and the latest explainability techniques?</i>

Responsible AI (RAI)

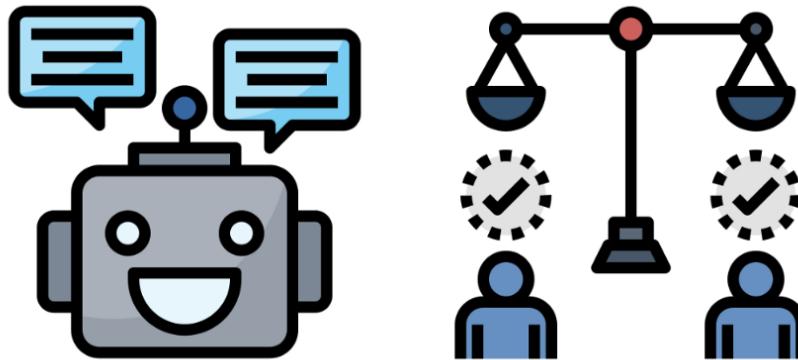
- ▶ Responsible AI principles (from Microsoft):
 - ▶ **Fairness:** AI systems should treat all people fairly
 - ▶ **Reliability & Safety:** AI systems should perform reliably and safely
 - ▶ **Privacy & Security:** AI systems should be secure and respect privacy
 - ▶ **Inclusiveness:** AI systems should empower everyone and engage people
 - ▶ **Transparency:** AI systems should be understandable
 - ▶ **Accountability:** People should be accountable for AI systems

Responsible AI (RAI)

- ▶ Responsible AI principles (from Google):

- ▶ Fairness
- ▶ Interpretability
- ▶ Privacy
- ▶ Security

Fairness and Interpretability



- ▶ Interpretability: understanding how models make predictions.
- ▶ Fairness: understanding if predictions are biased towards certain groups.
- ▶ Interpretability does not necessarily imply fairness, yet an interpretable model is still more likely to be fair.



Introduction to Fairness



Fairness Concerns

Amazon ditched AI recruiting tool that favored men for technical jobs

Specialists had been building computer programs since 2014 to review résumés in an effort to automate the search process



Amazon's automated hiring tool was found to be inadequate after penalizing the résumés of female candidates. Photograph: Brian Snyder/Reuters

The data on which the AI hiring algorithm was trained created a preference for male candidates.¹⁾

RESEARCH ARTICLE

ECONOMICS

Dissecting racial bias in an algorithm used to manage the health of populations

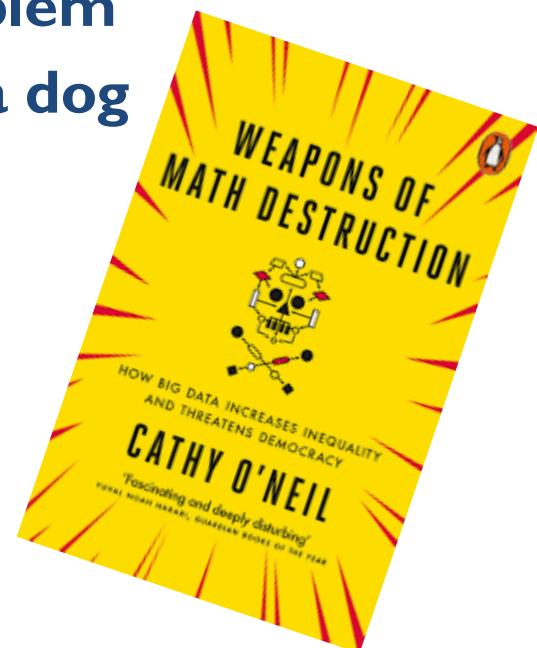
Ziad Obermeyer^{1,2*}, Brian Powers³, Christine Vogeli⁴, Sendhil Mullainathan^{5*†}

Health systems rely on commercial prediction algorithms to identify and help patients with complex health needs. We show that a widely used algorithm, typical of this industry-wide approach and affecting millions of patients, exhibits significant racial bias: At a given risk score, Black patients are considerably sicker than White patients, as evidenced by signs of uncontrolled illnesses. Remedy this disparity would increase the percentage of Black patients receiving additional help from 17.7 to 46.5%. The bias arises because the algorithm predicts health care costs rather than illness, but unequal access to care means that we spend less money caring for Black patients than for White patients. Thus, despite health care cost appearing to be an effective proxy for health by some measures of predictive accuracy, large racial biases arise. We suggest that the choice of convenient, seemingly effective proxies for ground truth can be an important source of algorithmic bias in many contexts.

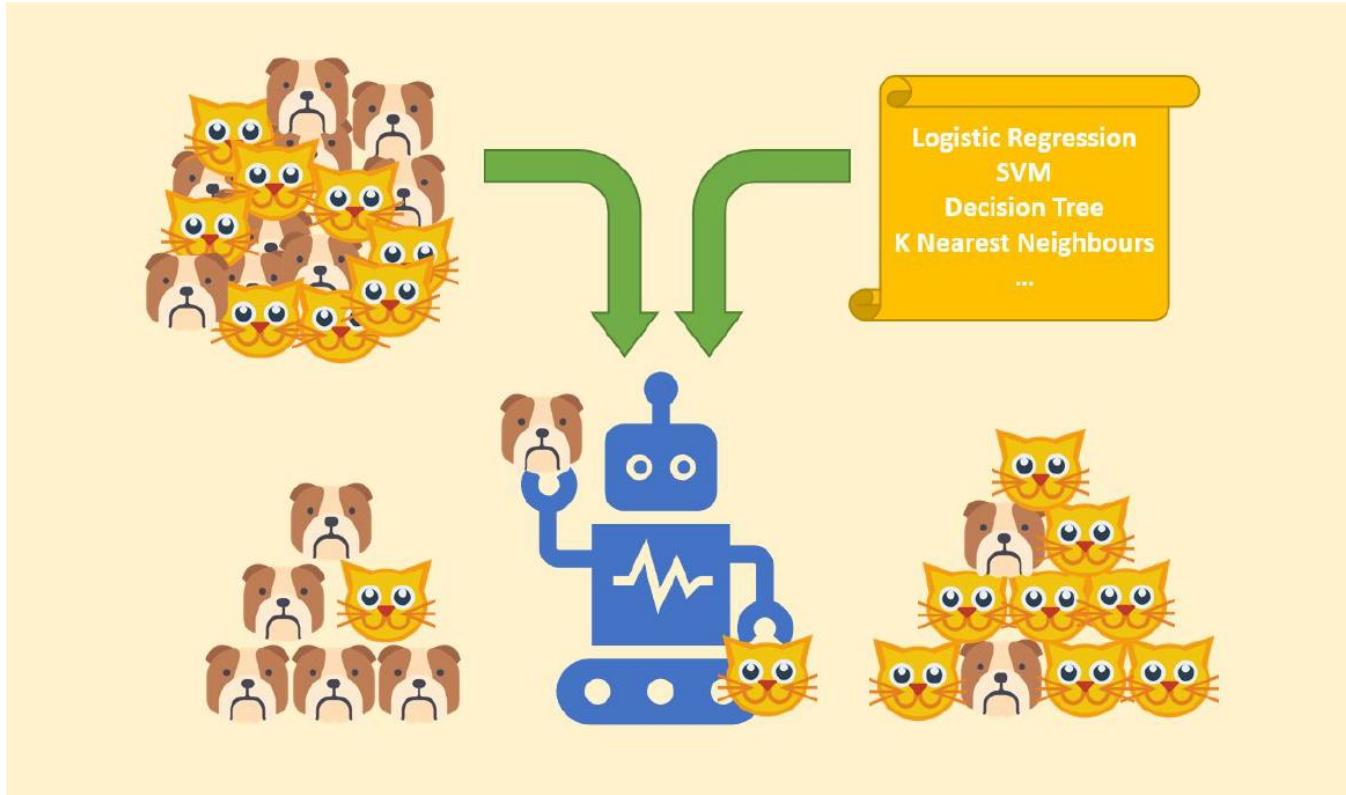
Industry-wide approach affecting millions of patients exhibits significant racial bias.²⁾

Data Science

- ▶ Input: Data
- ▶ Process data through some algorithm (regression, neural networks, SVM, etc.)
 - ▶ The algorithm learns from the data you feed it
- ▶ Output: a decision/**prediction**
 - ▶ **Often, it is posed as a classification problem**
 - ▶ e.g., predict if an image is a dog or not a dog



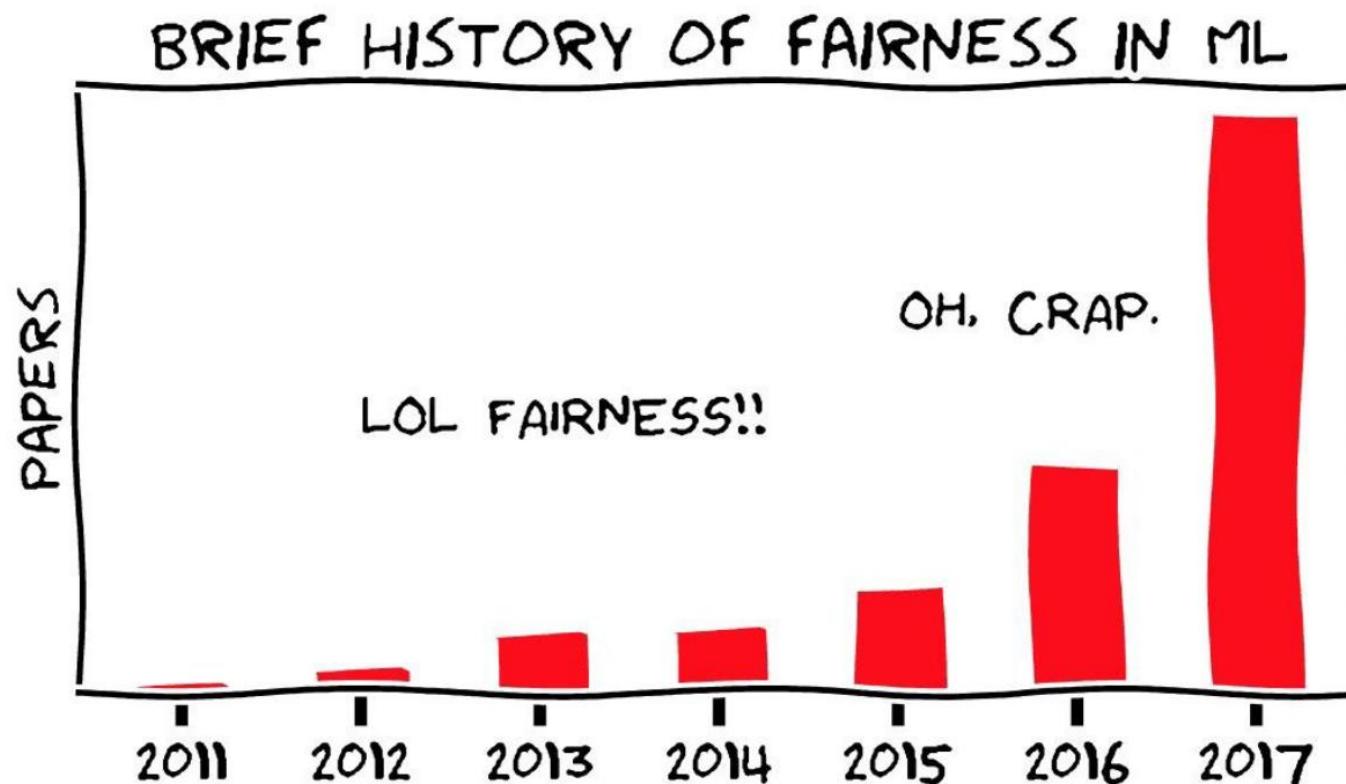
Classification Task



Algorithmic Fairness

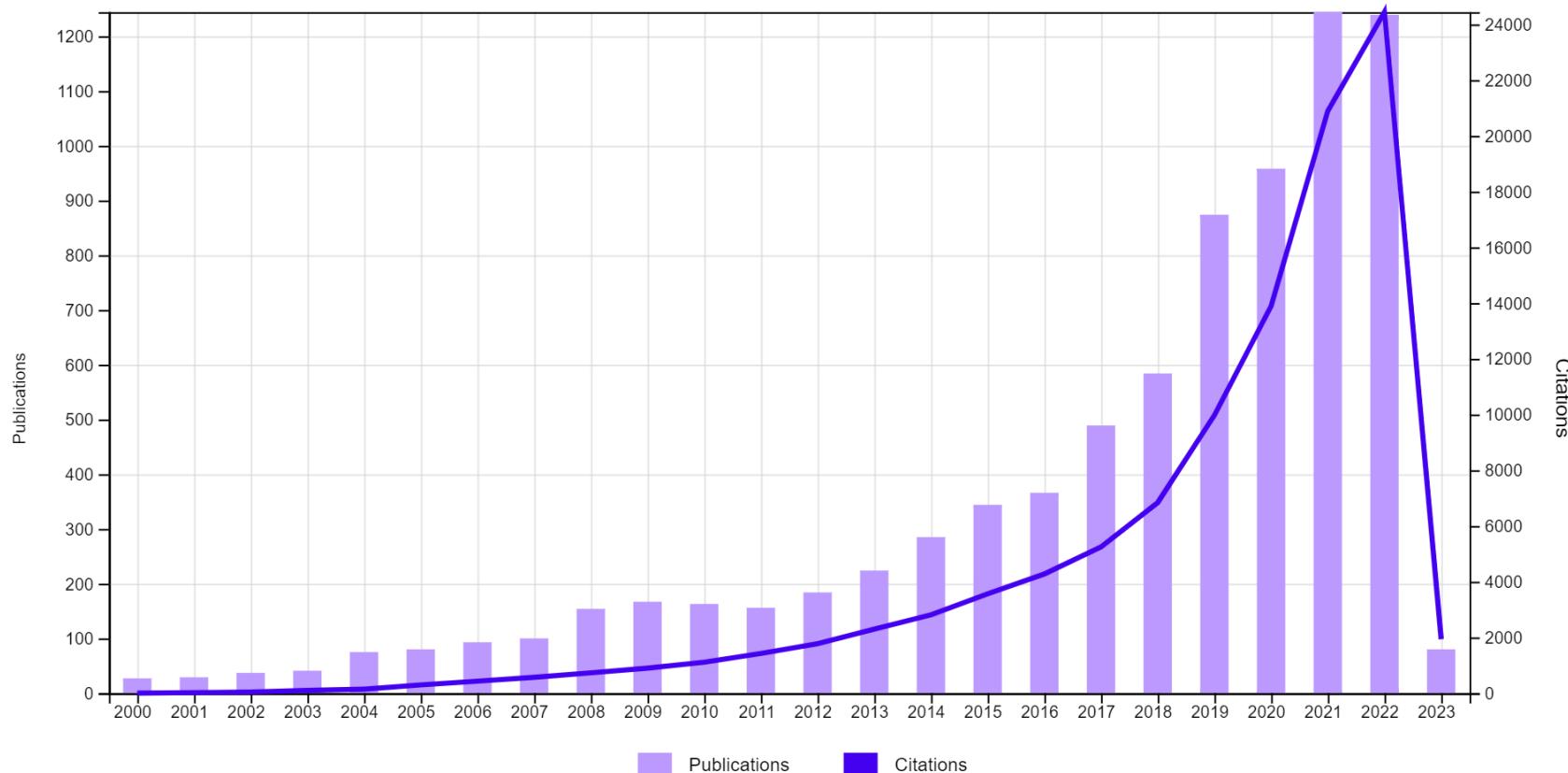
- ▶ A population is diverse: race, religion, geographic location, gender, etc.
- ▶ However, different demographic groups have different unfairness they experience.

Fairness in ML



Fairness in ML

- ▶ Web of Science search results with “fairness machine learning” OR “fairness data science” OR “fairness artificial intelligence”

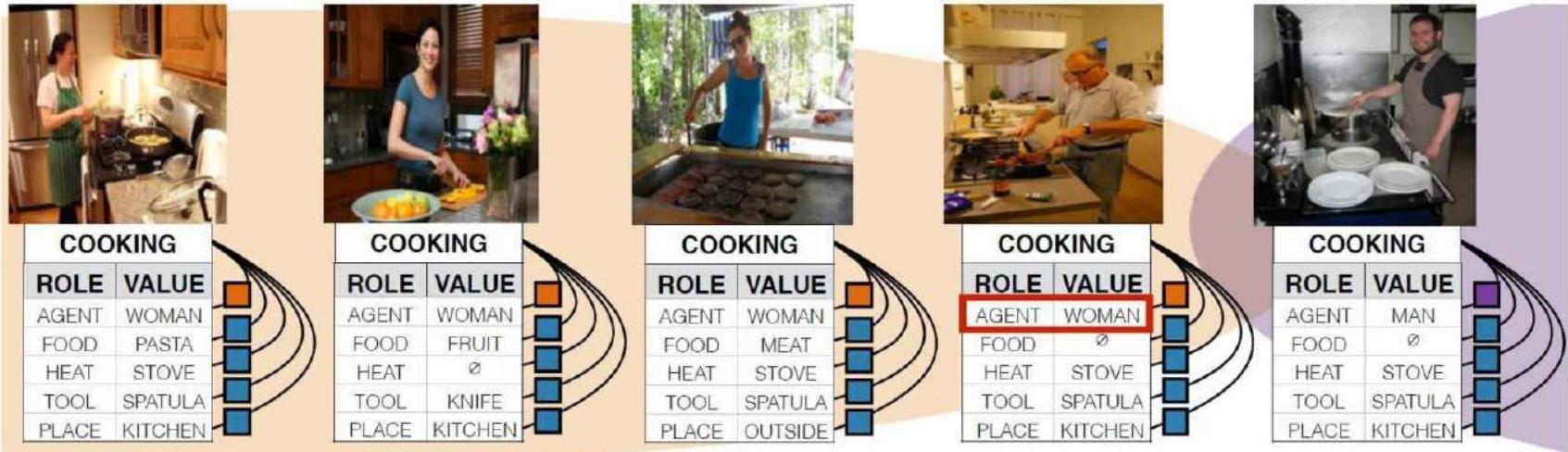


Why are algorithms unfair?

- ▶ Training data is unrepresentative
 - ▶ Data is accumulated over time – historical biases
 - ▶ Data is gathered/labelled by people – societal biases
- ▶ Sometimes, features can serve as proxies for others
 - ▶ Zip code (location) and race

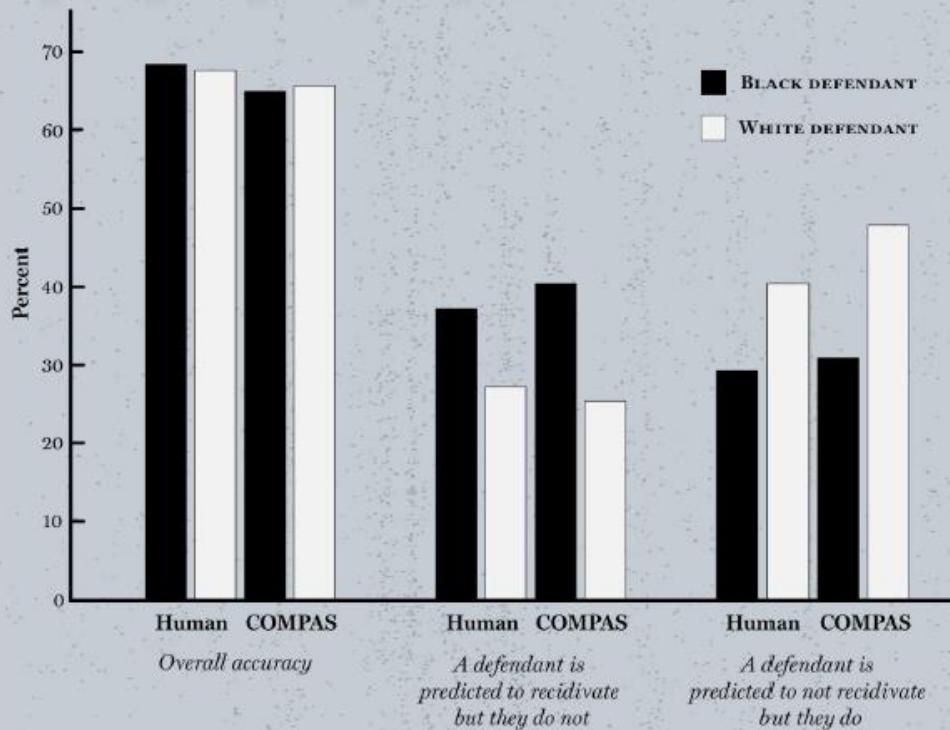
Machine Learning can amplify bias.

Men Also Like Shopping:
Reducing Gender Bias Amplification using Corpus-level Constraints



- Data set: 67% of people cooking are women
- Algorithm predicts: 84% of people cooking are women

COMMERCIAL SOFTWARE NO MORE ACCURATE THAN UNTRAINED PEOPLE IN PREDICTING RECIDIVISM



Participants saw a description of a defendant that did not include their race and predicted whether each individual would recidivate within 2 years of their most recent crime.

Here, human predictions are compared to COMPAS algorithmic predictions. Human participants responding to an online survey, presumably none of them criminal justice experts, were approximately as accurate as COMPAS, the new *Science Advances* study reveals.

Food for thought

- ▶ How do we (mathematically) define what it means for an algorithm to be fair?
- ▶ How do we use these definitions to construct algorithms that are fair?
- ▶ How do these algorithms impact all populations and subgroups? Who is affected?

Food for thought

- ▶ Who designed and created these algorithms?
- ▶ How do we teach future generations, who will use these algorithms, to think about these ethical considerations?
- ▶ How can we work together to make data science more transparent, accountable, and fair?

“ The Achilles’ heel of all algorithms is the humans who build them and the choices they make about outcomes, candidate predictors for the algorithm to consider, and the training sample...

**Algorithms change the landscape —
they do not eliminate the problem.**

— “Discrimination in the Age
of Algorithms”





Introduction to Interpretability



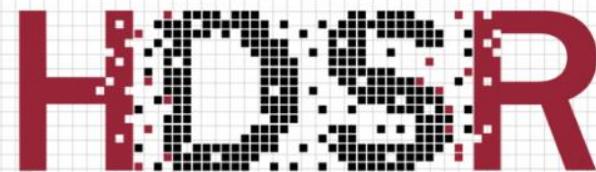
The Need for Interpretability



'Creative ... motivating' and fired



Sarah Wysocki was out of work for only a few days after she was fired by DCPS last year. She is now teaching at Hybla Valley Elementary School in Fairfax County. (Jahi Chikwendiu/The Washington Post)



HARVARD DATA SCIENCE REVIEW

Why Are We Using Black Box Models in AI When We Don't Need To? A Lesson From an Explainable AI Competition

by Cynthia Rudin and Joanna Radin

Published on Nov 22, 2019



TOPICS

RESEARCH

BLOG

PODCAST

ABOUT

BLOG 05 SEPTEMBER 2019

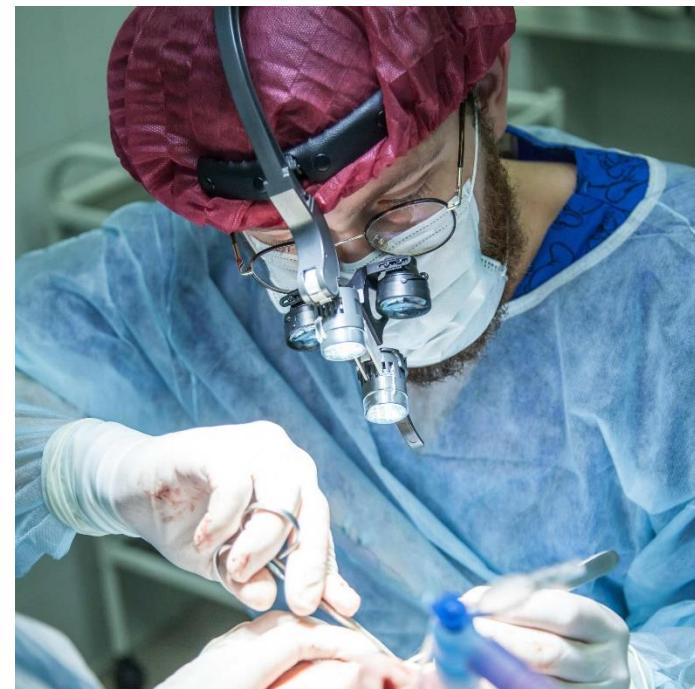
Algorithm Bias in Credit Scoring: What's Inside the Black Box?

Food for Thought

Assume that you have cancer and need surgery to remove a tumor.
Which one do you prefer?

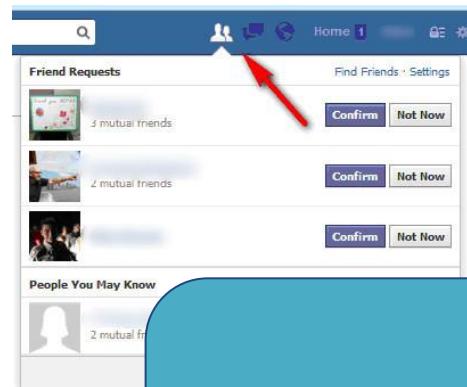


Robotic Arm
No questions asked
Failure Rate: 2%



Experienced Surgeon
Explain anything about the surgery
Failure Rate: 15%

Motivation



Amazon.com: Bestselling Canon Cameras [Newsletters](#) | X

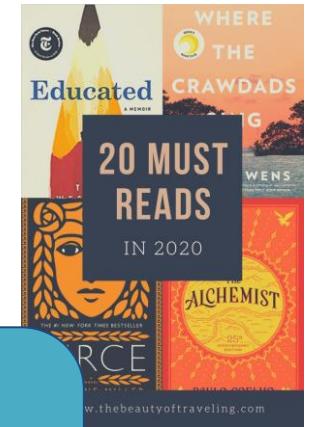
show details May 30 (9 days ago) [Reply](#)

Amazon.com to me

amazon.com More to Explore

Customers who have shown an interest in point-and-shoot cameras might like to see this week's bestselling models.

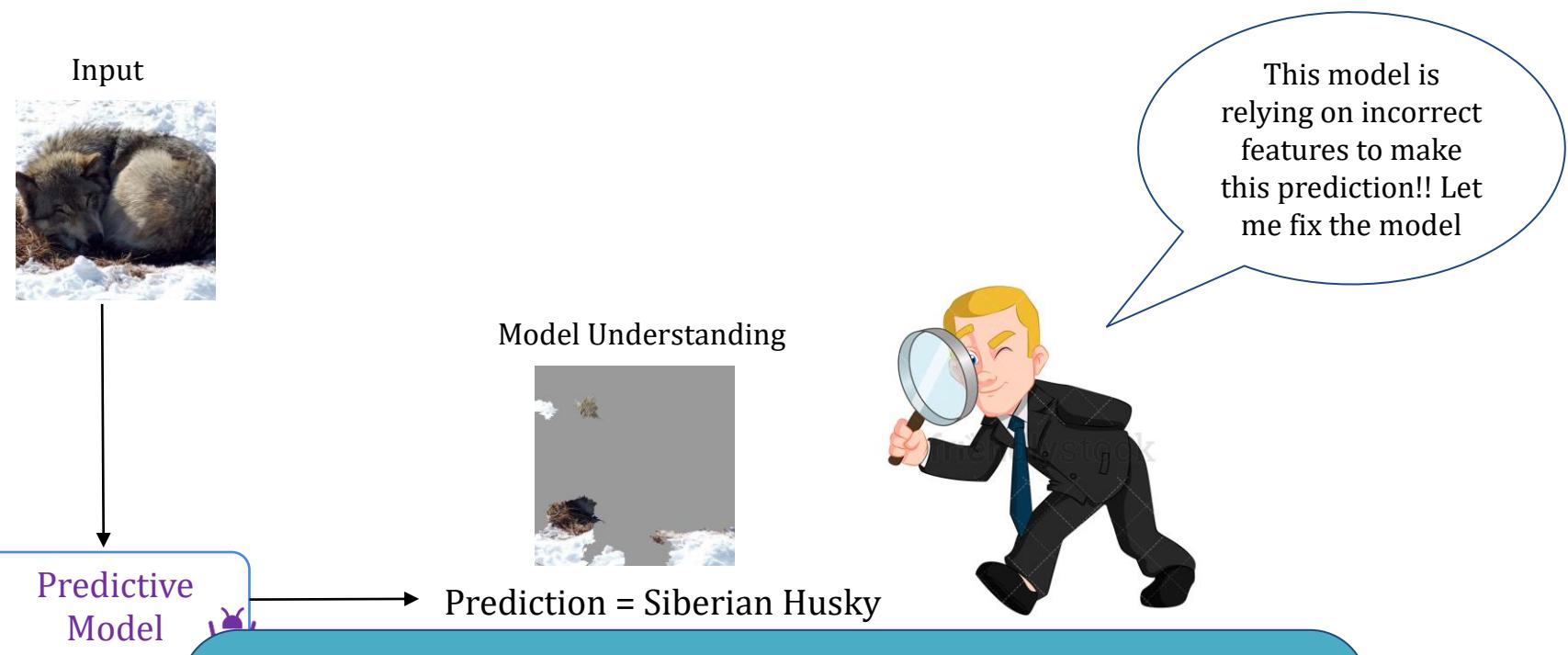
Canon PowerShot Canon PowerShot Canon PowerShot Canon PowerShot



Machine Learning is EVERYWHERE!!

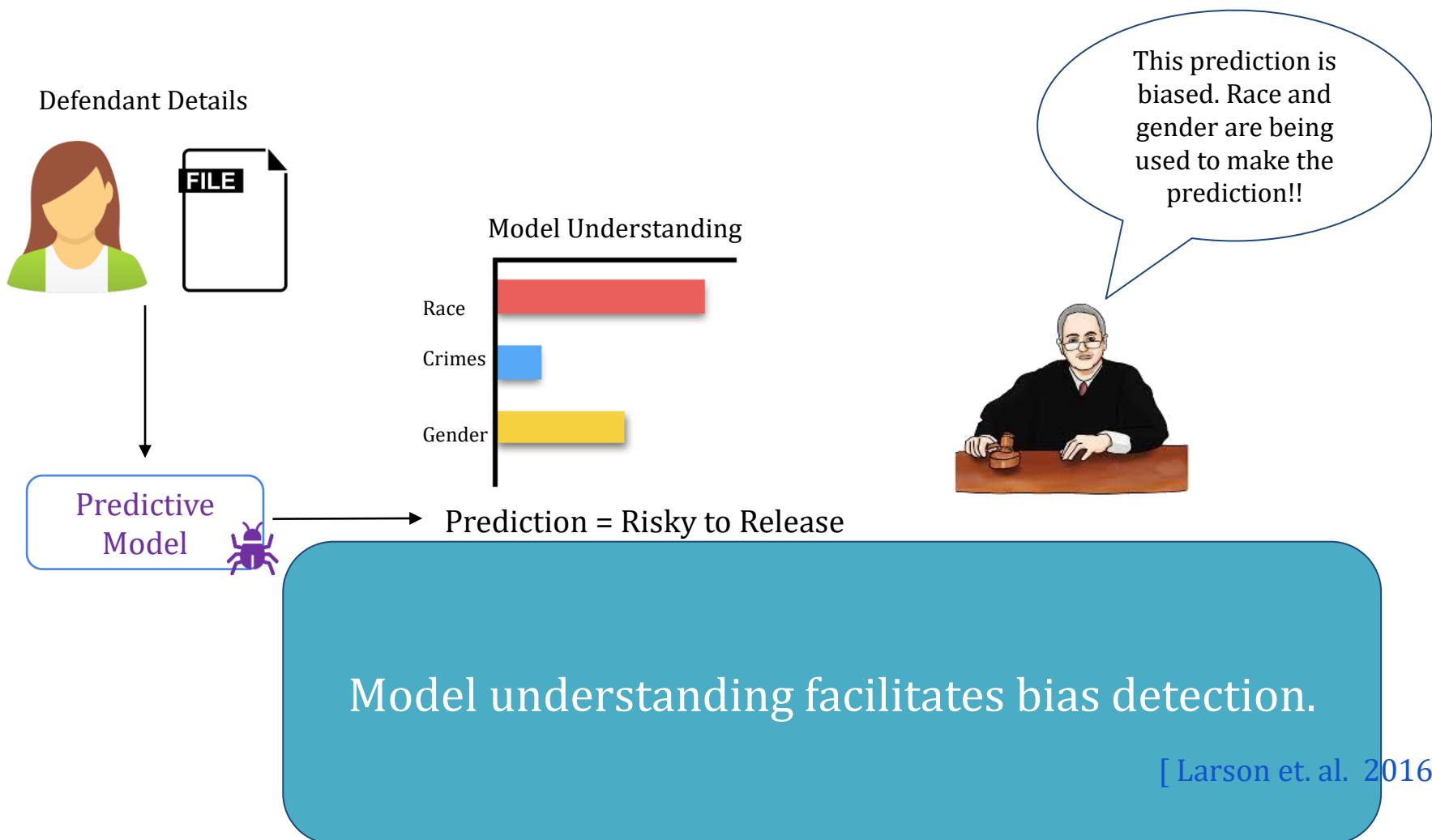
[Weller 2017]

Motivation: Why Model Understanding?

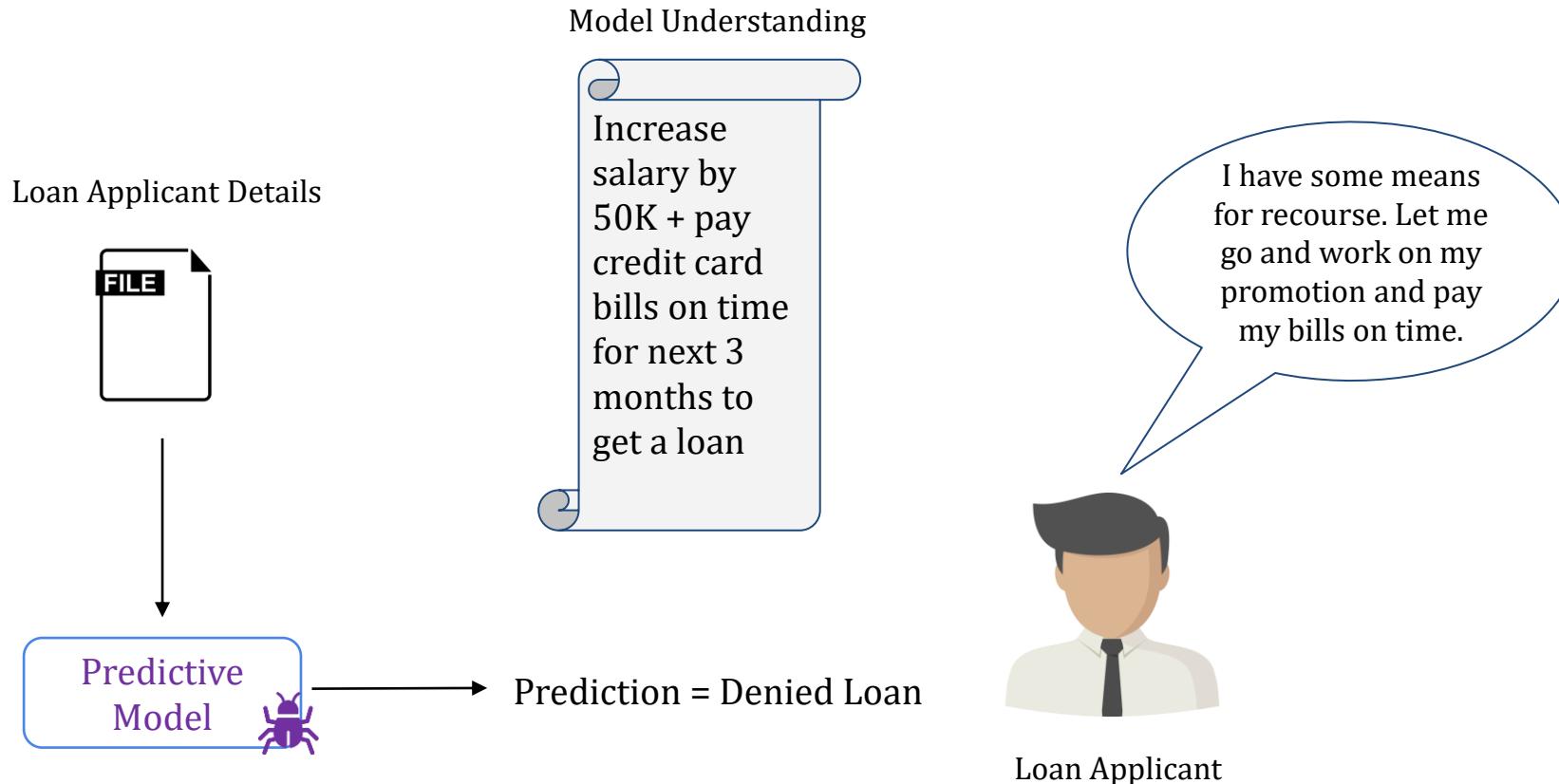


Model understanding facilitates debugging.

Motivation: Why Model Understanding?

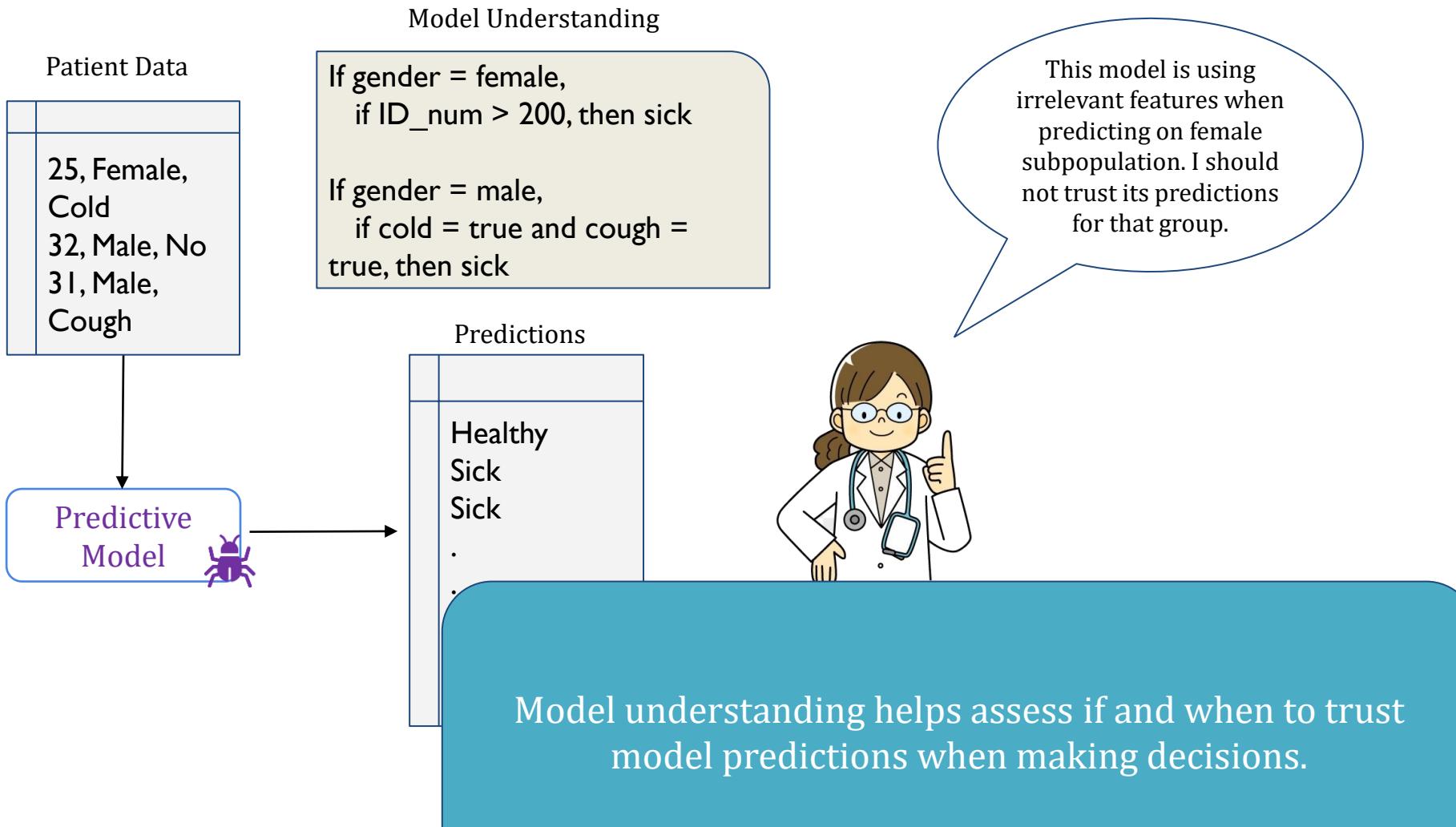


Motivation: Why Model Understanding?

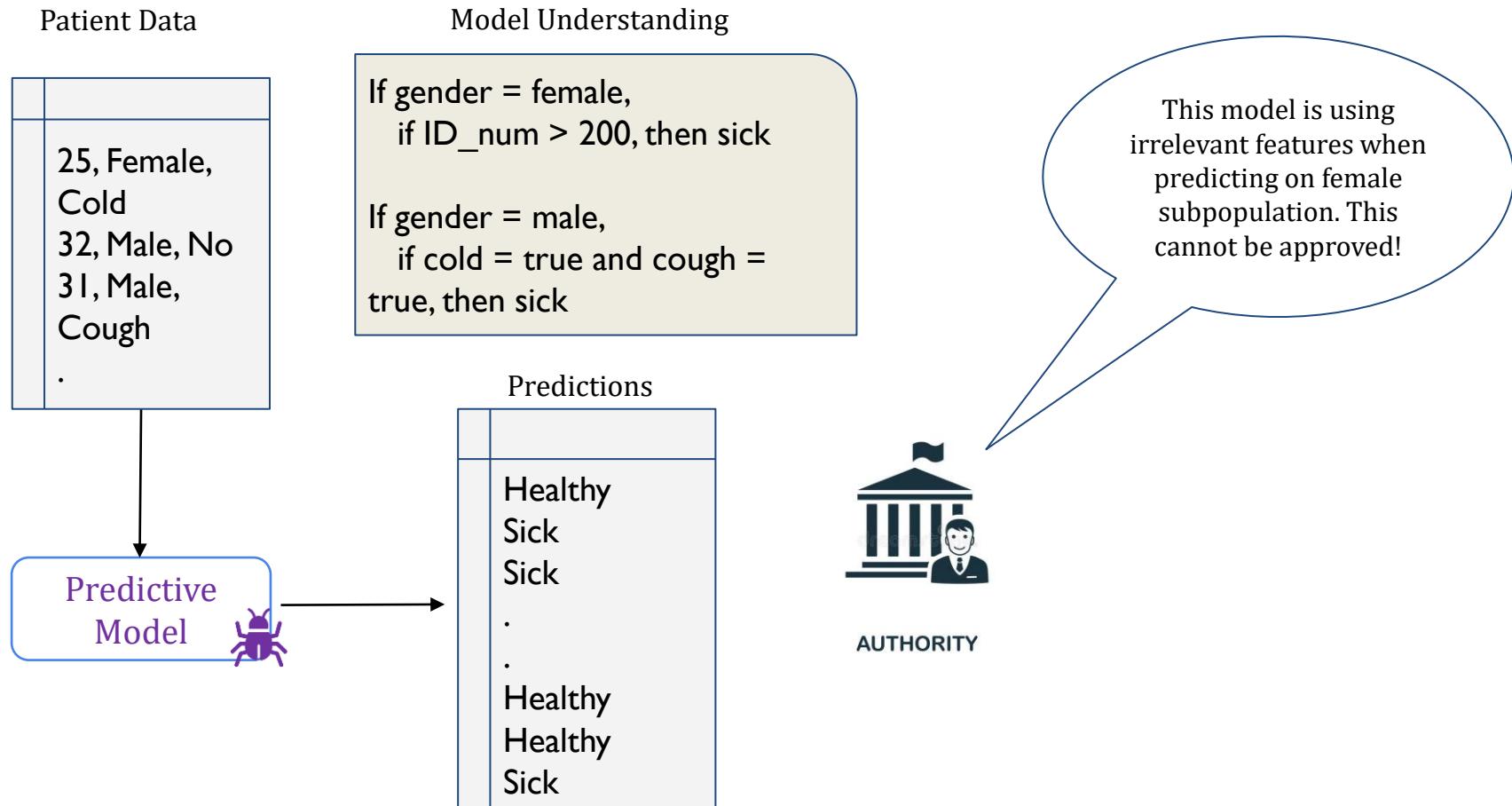


Model understanding helps provide recourse to individuals who are adversely affected by model predictions.

Motivation: Why Model Understanding?



Motivation: Why Model Understanding?



Motivation: Why Model Understanding?

Why do we want Interpretability?

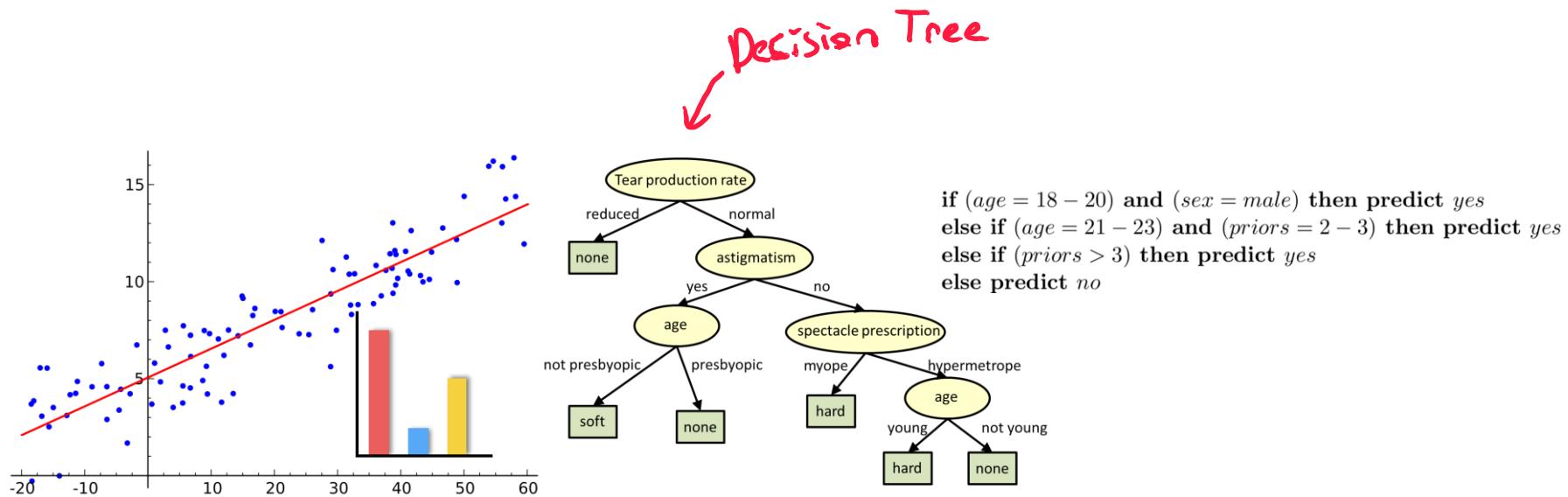
- Debugging
- Bias Detection
- Recourse
- If and when to trust model predictions
- Vet models to assess suitability for deployment

Stakeholders

- End users (e.g., loan applicants)
- Decision makers (e.g., doctors, judges)
- Regulatory agencies (e.g., FDA, European commission)
- Researchers and engineers

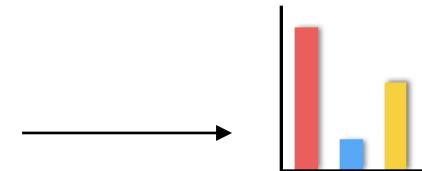
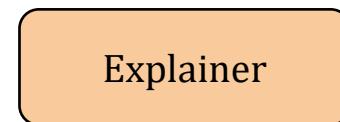
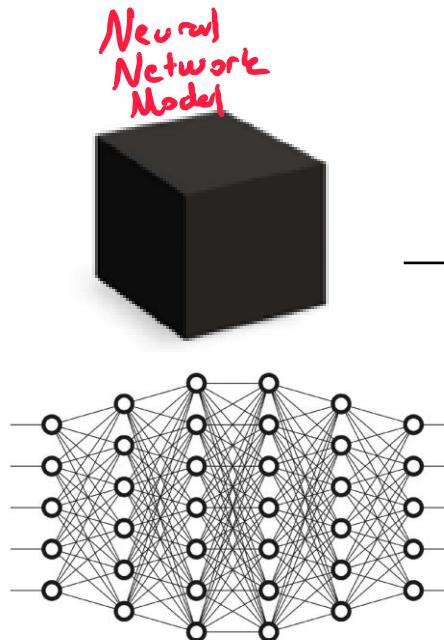
Achieving Model Understanding

Take I: Build *inherently interpretable* predictive models



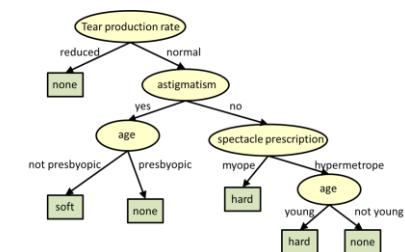
Achieving Model Understanding

Take 2: Explain pre-built models in a post-hoc manner



```
if (age = 18 – 20) and (sex = male) then predict yes  
else if (age = 21 – 23) and (priors = 2 – 3) then predict yes  
else if (priors > 3) then predict yes  
else predict no
```

Decision Tree



High-Stakes Decisions



- ▶ **Healthcare:** What **treatment** to recommend to the patient?
- ▶ **Criminal Justice:** Should the defendant be released on **bail**?

High-Stakes Decisions: Impact on human well-being.

Case Study: Evaluating an Interpretable System

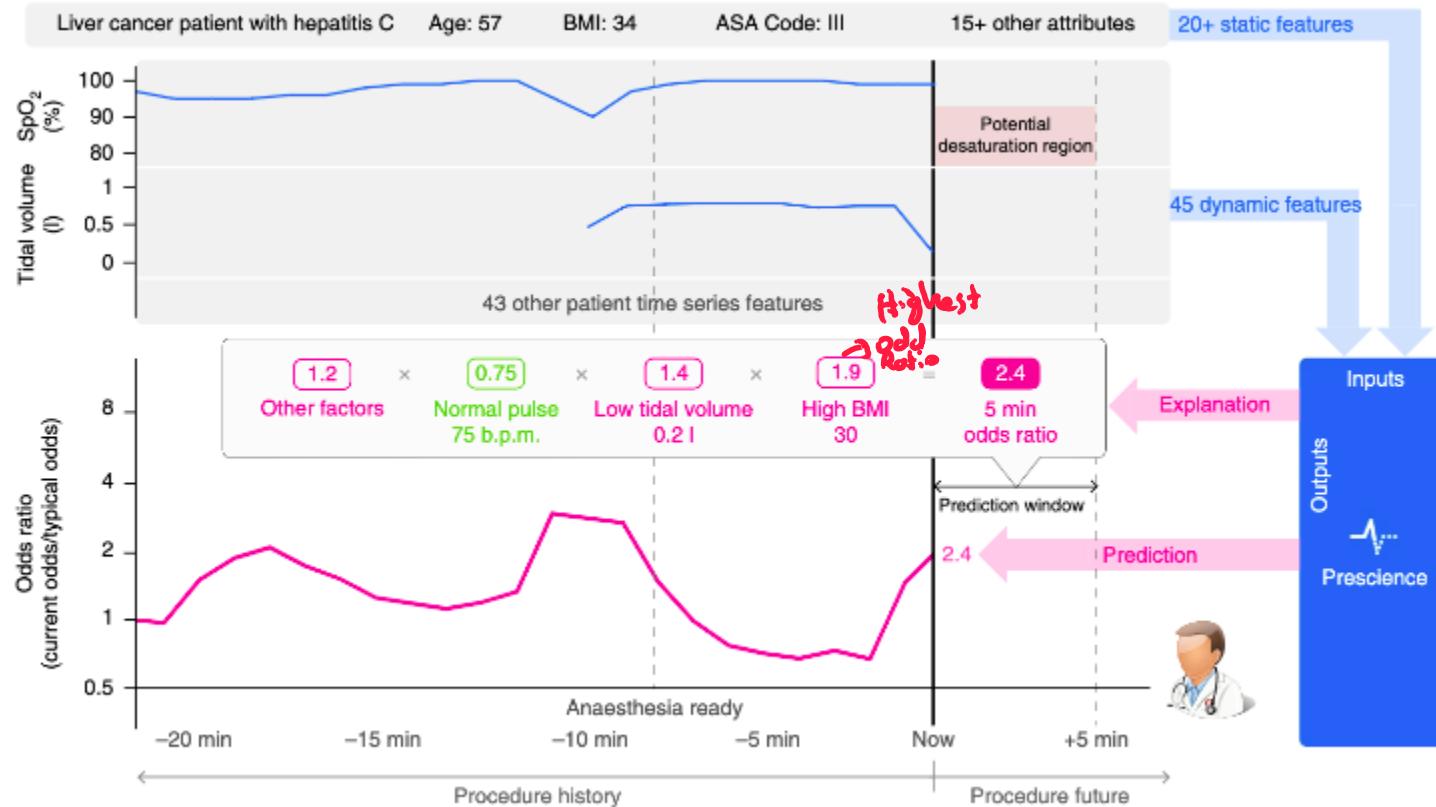
Explainable machine-learning predictions for the prevention of hypoxaemia during surgery

Scott M. Lundberg¹, Bala Nair^{2,3,4}, Monica S. Vavilala^{2,3,4}, Mayumi Horibe⁵, Michael J. Eisses^{2,6}, Trevor Adams^{2,6}, David E. Liston^{2,6}, Daniel King-Wai Low^{2,6}, Shu-Fang Newman^{2,3}, Jerry Kim^{2,6} and Su-In Lee^{1*}

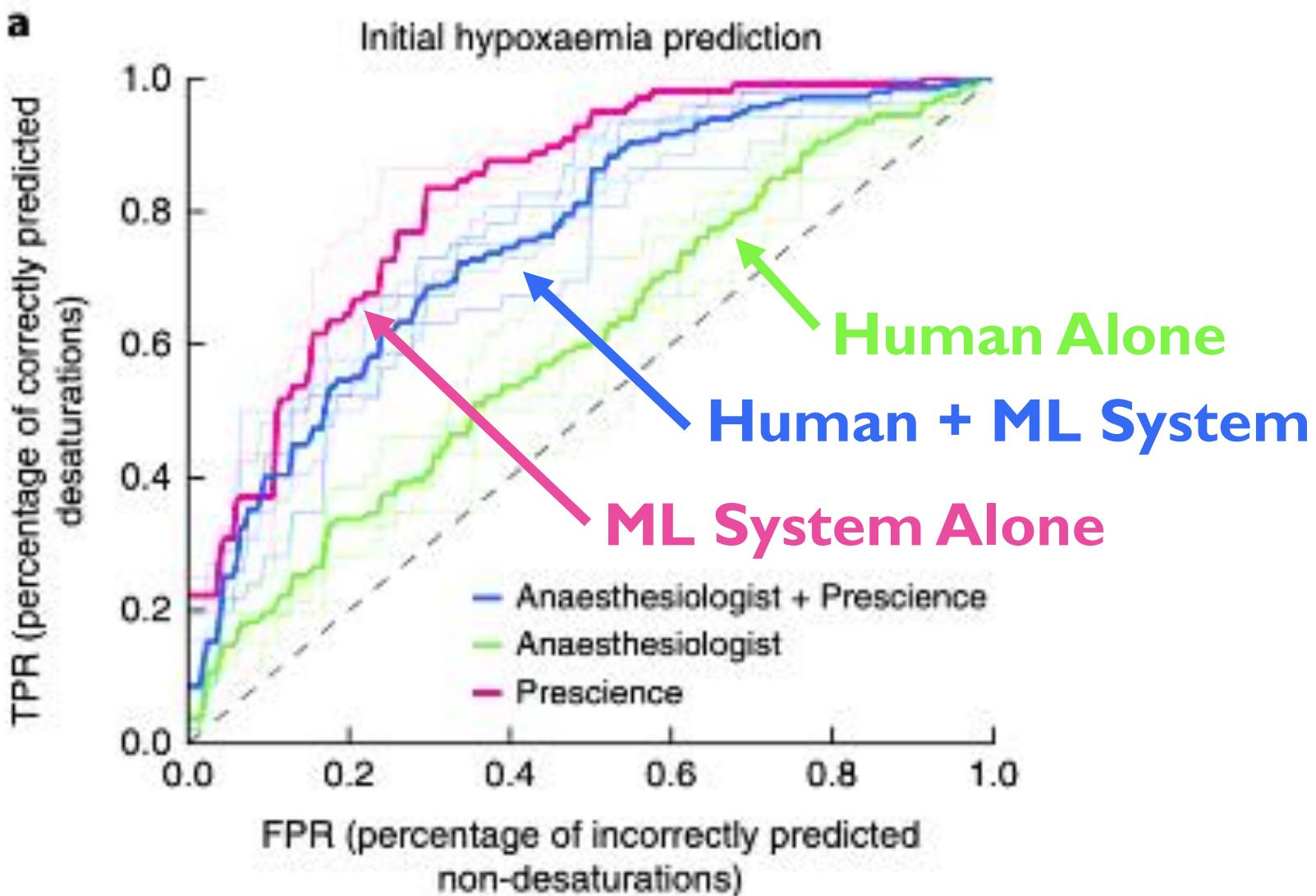
Case Study: Evaluating an Interpretable System

- ▶ Application: Preventing Hypoxaemia During Surgery
 - ▶ Low arterial blood oxygen tension
 - ▶ Good predictions would allow anesthesiologists to take preventative measures during surgery
- ▶ User study comparing predictive performance of humans and models

Case Study: Evaluating an Interpretable System

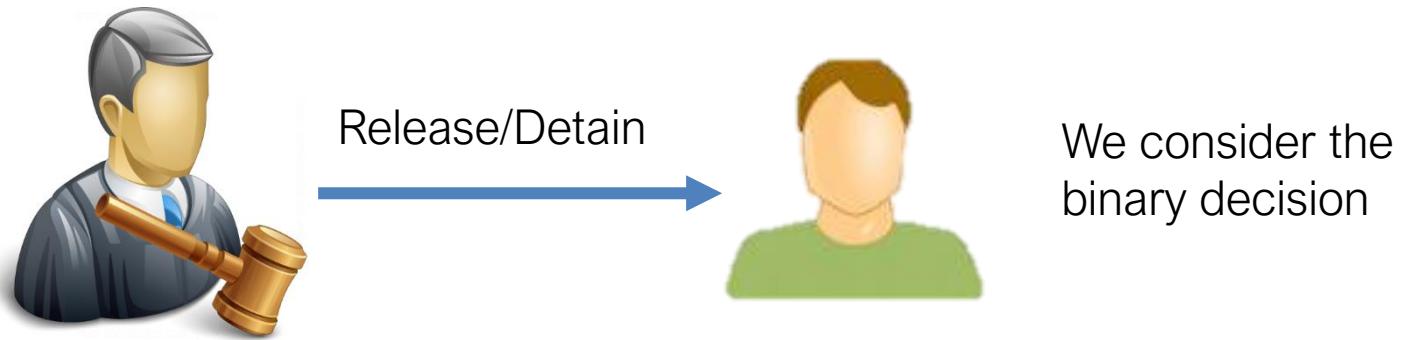


Case Study: Evaluating an Interpretable System



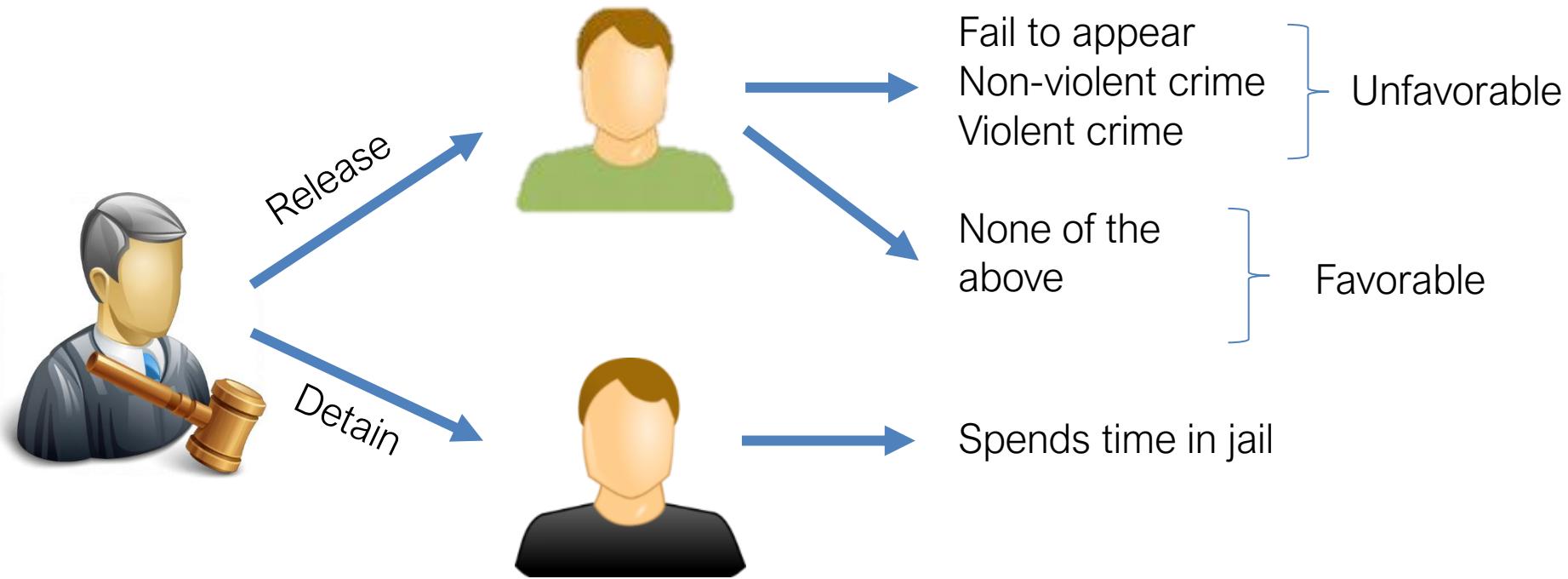
Real World Scenario: Bail Decision

- ▶ U.S. police make about 12M arrests each year



- ▶ Release vs. Detain is a high-stakes decision
 - ▶ Pre-trial detention can go up to 9 to 12 months
 - ▶ Consequential for jobs & families of defendants as well as crime

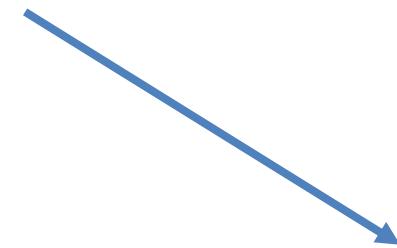
Bail Decision



Judge is making a prediction:
Will the defendant commit 'crime' if released on bail?

Bail Decision-Making as a Prediction Problem

Does making the model more understandable/transparent to the judge improve decision-making performance?
If so, how to do it?



Predictive
Model

User Experiment

If Current-Offense = Felony:

- If Prior-Felony = Yes and Prior-Arrests ≥ 1 , then Crime
- If Crime-Status = Active and Owns-House = No and Has-Kids = No, then Crime
- If Prior-Convictions = 0 and College = Yes and Owns-House = Yes, then No Crime

If Current-Offense = Misdemeanor and Prior-Arrests > 1 :

- If Prior-Jail-Incarcerations = Yes, then Crime
- If Has-Kids = Yes and Married = Yes and Owns-House = Yes, then No Crime
- If Lives-with-Partner = Yes and College = Yes and Pays-Rent = Yes, then No Crime

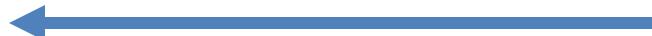
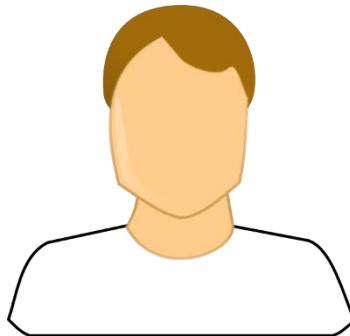
If Current-Offense = Misdemeanor and Prior-Arrests ≤ 1 :

- If Has-Kids = No and Owns-House = No and Moved_10times_5years = Yes, then Crime
- If Age ≥ 50 and Has-Kids = Yes, then No Crime

Default: No Crime

Judges were able to make decisions 2.8 times faster and 38% more accurately (compared to no explanation and only prediction) !

Real World Scenario: Treatment Recommendation



Demographics:

Age

Gender

.....

Medical History:

Has asthma?

Other chronic issues?

.....

Symptoms:

Severe Cough

Wheezing

.....

Test Results:

Peak flow: Positive

Spirometry: Negative

What treatment should be given?
Options: quick relief drugs (mild),
controller drugs (strong)

Treatment Recommendation



Symptoms relieved in

User studies showed that doctors were able to make decisions 1.9 times faster and 26% more accurately when explanations were provided along with the model!



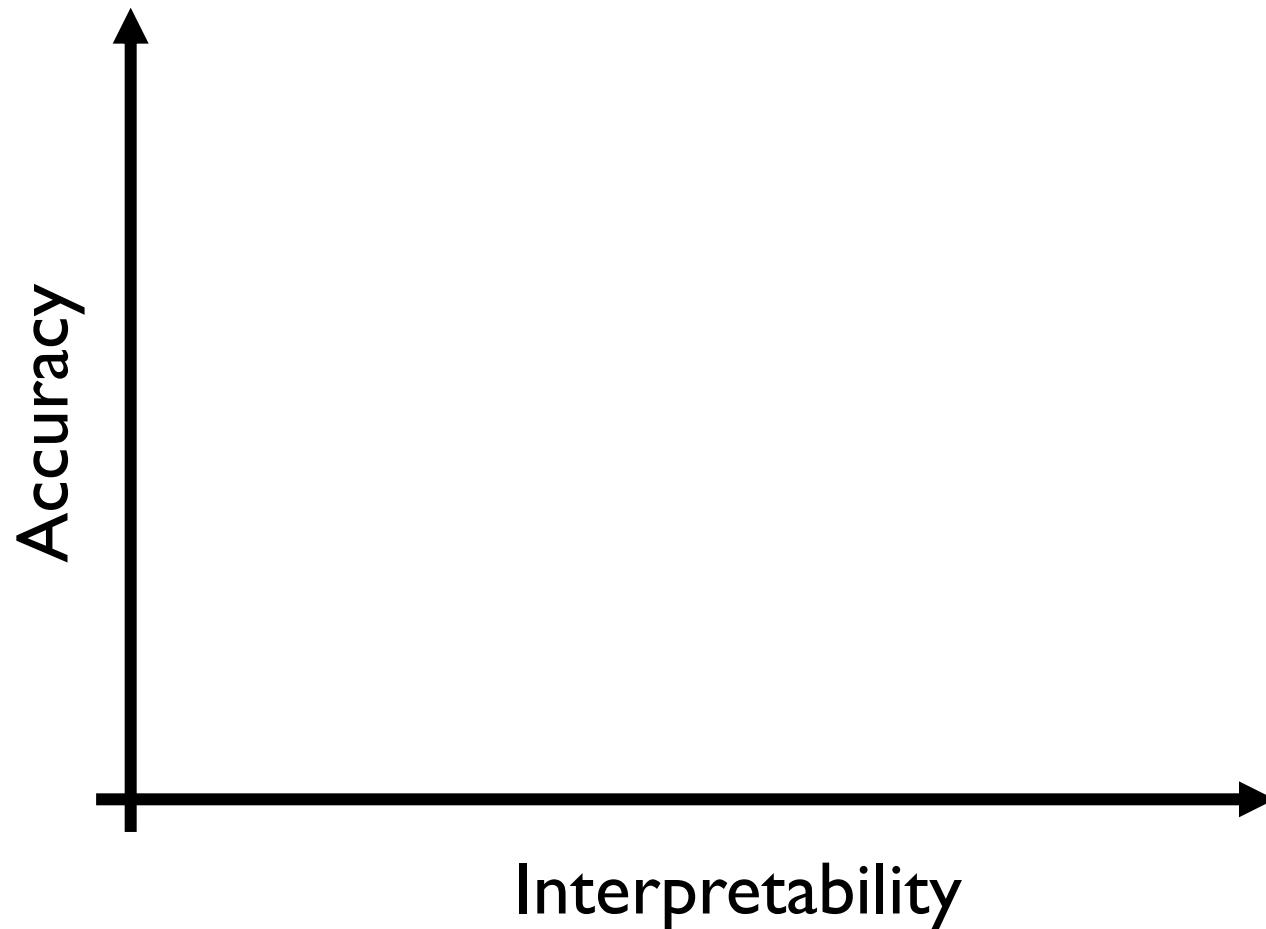
strong



Symptoms relieved in
➤ Within a week }

Doctor is making a prediction:
Will the patient get better with a milder drug?
Use ML to make a similar prediction

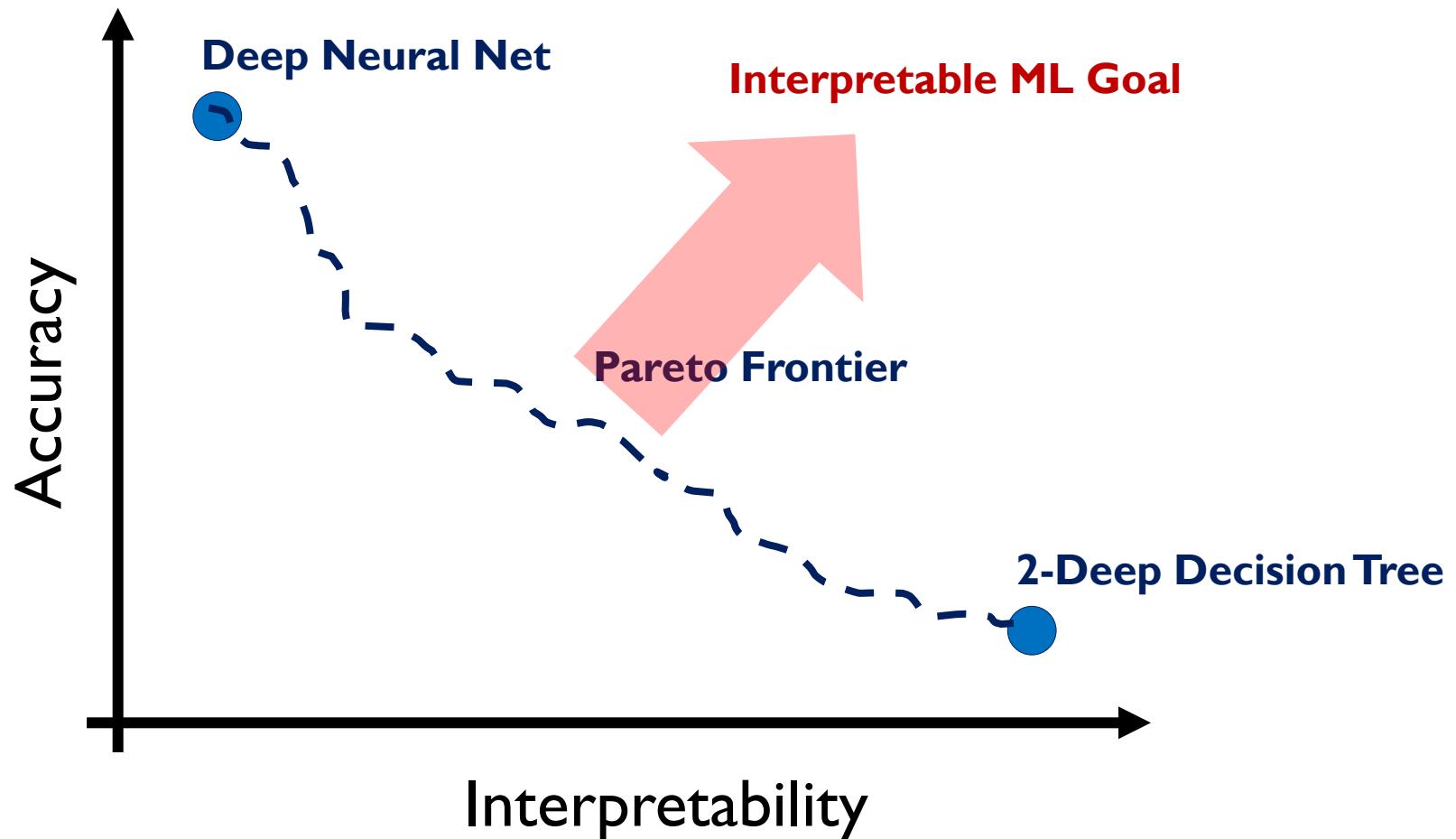
Interpretability vs Accuracy



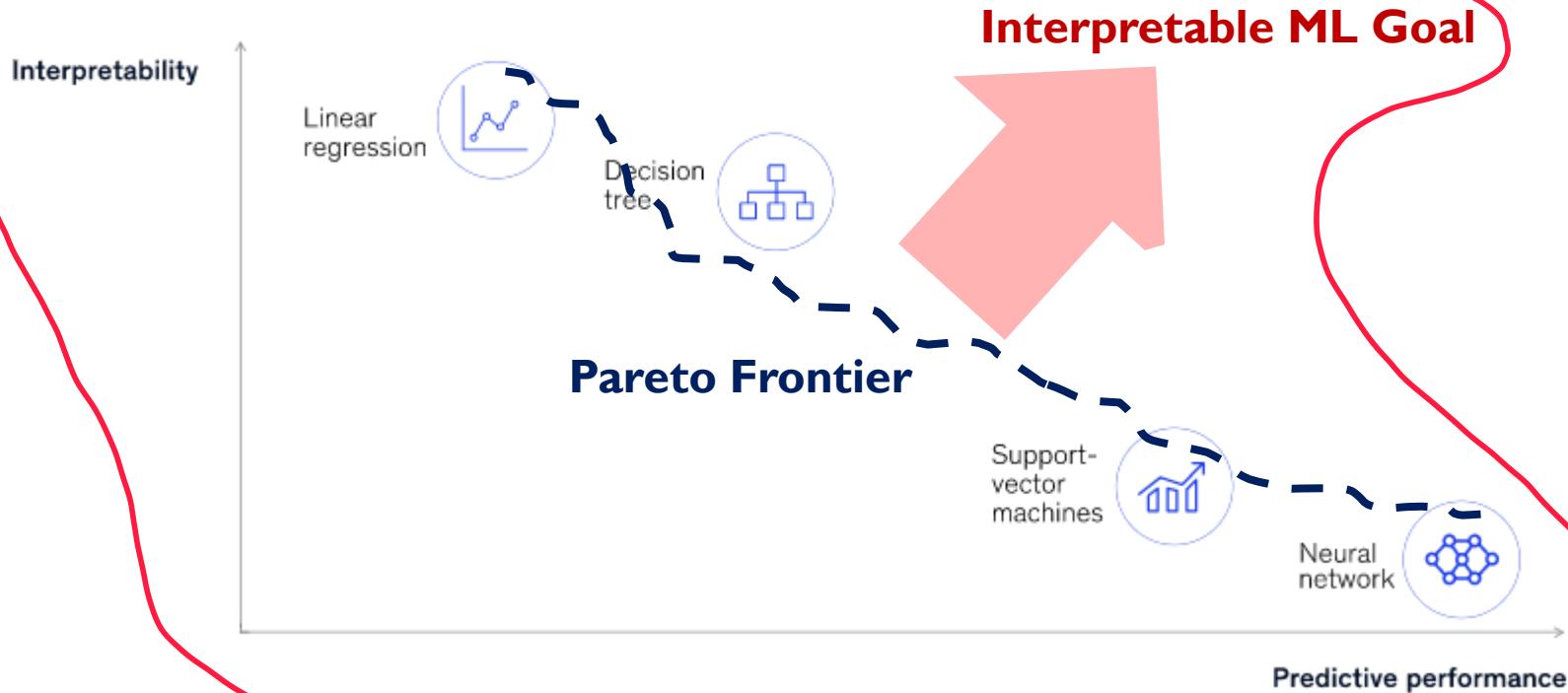
How to interpret specific models

- ▶ Are there other methods that are easier to interpret/understand out of the box?
 - ▶ Decision Trees
 - ▶ KNN
 - ▶ SVMs
 - ▶ RFs
 - ▶ NNs

Interpretability vs Accuracy

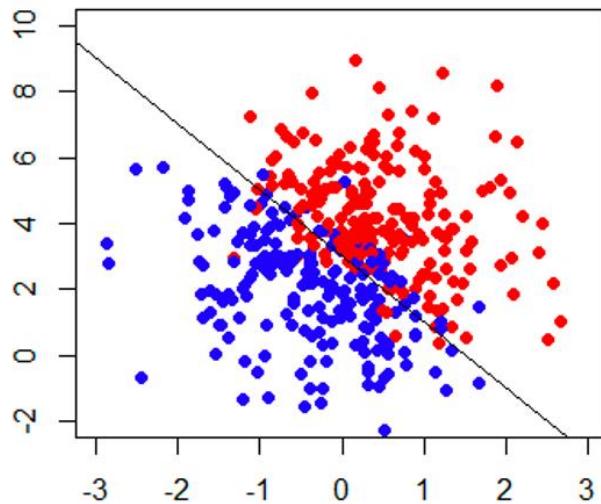


Interpretability vs Accuracy

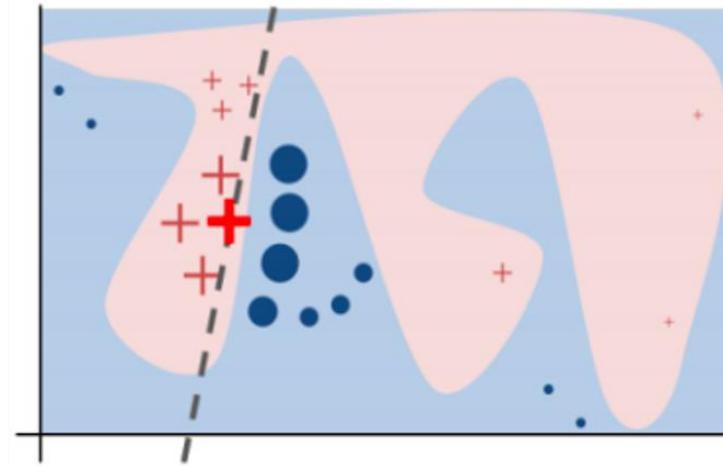


In **certain** settings, accuracy-interpretability trade offs may exist.

Inherently Interpretable Models vs. Post hoc Explanations



can build interpretable +
accurate models



complex models might
achieve higher accuracy

Inherently Interpretable Models vs. Post hoc Explanations

Sometimes, you don't have enough data to build your model from scratch.

And, all you have is a (proprietary) black box!



Inherently Interpretable Models vs. Post hoc Explanations

If you *can build* an interpretable model which is also adequately accurate for your setting, DO IT!

Otherwise, *post hoc explanations* come to the rescue!



Let's get into some details!

Next Up!

- ▶ Define and evaluate interpretability
 - ▶ somewhat! 😊
- ▶ Taxonomy of interpretability evaluation
- ▶ Taxonomy of interpretability based on applications/tasks
- ▶ Taxonomy of interpretability based on methods

Motivation for Interpretability

- ▶ ML systems are being deployed in complex high-stakes settings
- ▶ Accuracy alone is no longer enough
- ▶ Auxiliary criteria are important:
 - ▶ Safety
 - ▶ Nondiscrimination
 - ▶ Right to explanation

Motivation for Interpretability

- ▶ Auxiliary criteria are often **hard to quantify** (completely)
 - ▶ E.g.: Impossible to enumerate all scenarios violating safety of an autonomous car
- ▶ Fallback option: interpretability
 - ▶ *If the system can explain its reasoning, we can verify if that reasoning is sound w.r.t. auxiliary criteria*

Prior Work: Defining and Measuring Interpretability

- ▶ Little consensus on what interpretability is and how to evaluate it
- ▶ Interpretability evaluation typically falls into:
 - ▶ Evaluate in the context of an application
 - ▶ Evaluate via a quantifiable proxy

Prior Work: Defining and Measuring Interpretability

- ▶ Evaluate in the **context of an application**
 - ▶ If a system is useful in a practical application or a simplified version, it must be interpretable
- ▶ Evaluate via a **quantifiable proxy**
 - ▶ Claim some model class is interpretable and present algorithms to optimize within that class
 - ▶ E.g. rule lists

You will know it when you see it!

Lack of Rigor?

Important to formalize these notions!!!

- ▶ Are all models in all “interpretable” model classes equally interpretable?
 - ▶ Model sparsity allows for comparison
- ▶ How to compare a linear model with a decision tree?
- ▶ Do all applications have same interpretability needs?

What is Interpretability?

- Interpretability*
- ▶ **Defn:** Ability to explain or to present in understandable terms to a human
 - ▶ No clear answers in psychology to:
 - ▶ What constitutes an explanation?
 - ▶ What makes some explanations better than the others?
 - ▶ When are explanations sought?

When and Why Interpretability?

- ▶ Not all ML systems require interpretability
 - ▶ E.g., ad servers, postal code sorting
 - ▶ No human intervention
- ▶ No explanation needed because:
 - ▶ No consequences for unacceptable results
 - ▶ Problem is well studied and validated well in real-world applications → trust system's decision

When do we need explanation then?

When and Why Interpretability?

- ▶ **Incompleteness** in problem formalization
 - ▶ Hinders optimization and evaluation

- ▶ Incompleteness \neq Uncertainty
 - ▶ Uncertainty can be quantified
 - ▶ E.g., trying to learn from a small dataset (uncertainty)

Incompleteness: Illustrative Examples

▶ Scientific Knowledge

- ▶ E.g., understanding the characteristics of a large dataset
- ▶ Goal is abstract

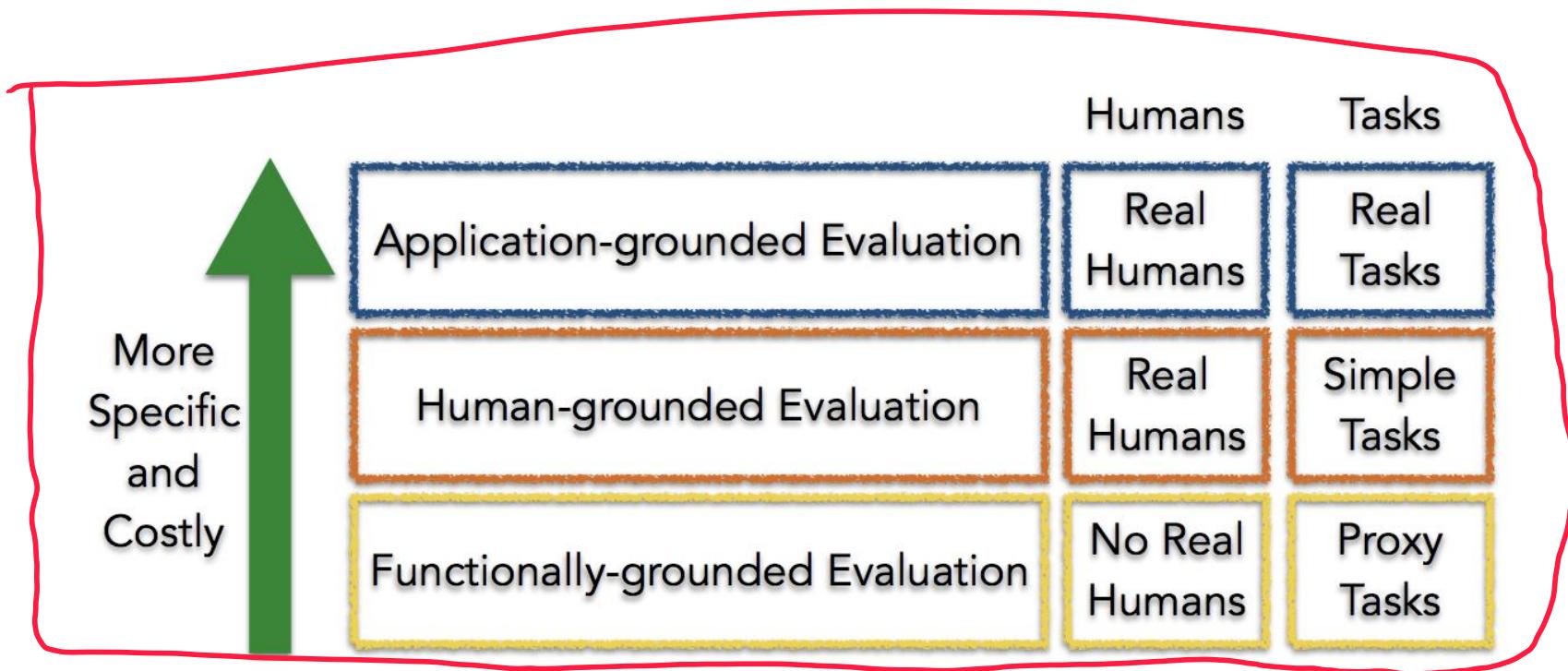
▶ Safety

- ▶ End to end system is never completely testable
- ▶ Not possible to check all possible inputs

▶ Ethics

- ▶ Guard against certain kinds of discrimination which are too abstract to be encoded
- ▶ No idea about the nature of discrimination beforehand

Taxonomy of Interpretability Evaluation



Claim of the research should match the type of the evaluation!

Application-grounded evaluation

- ▶ Real humans (domain experts), real tasks
- ▶ Domain experts experiment with exact application task
- ▶ Domain experts experiment with a simpler or partial task
 - ▶ Shorten experiment time
 - ▶ Increases number of potential subjects
- ▶ Typical in HCI and visualization communities

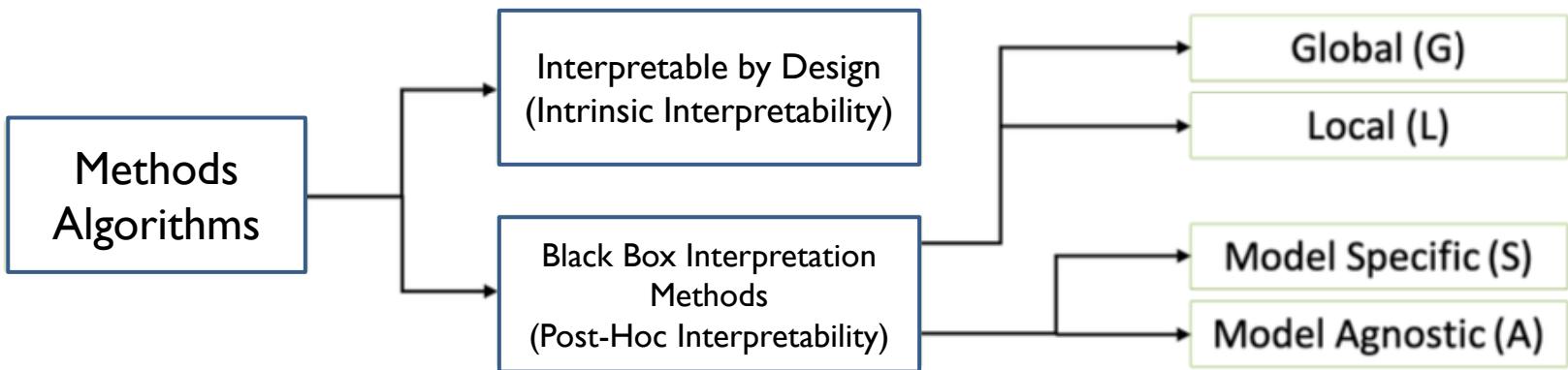
Human-grounded evaluation

- ▶ Real humans, simplified tasks
 - ▶ Can be completed with lay humans
 - ▶ Larger pool, less expensive
- ▶ Potential experiments
 - ▶ Pairwise comparisons
 - ▶ Simulate the model output
 - ▶ What changes should be made to input to change the output?

Functionally-grounded evaluation

- ▶ **No humans, just proxies**
 - ▶ Appropriate for a class of models already validated
 - ▶ E.g., decision trees
 - ▶ A method is not yet mature
 - ▶ Human subject experiments are unethical
 - ▶ What proxies to use?
- ▶ **Potential experiments**
 - ▶ Complexity (of a decision tree) compared to other other models of the same (similar) class
 - ▶ How many levels? How many rules?

Many Approaches to Interpretability



ÖZYEĞİN ÜNİVERSİTESİ

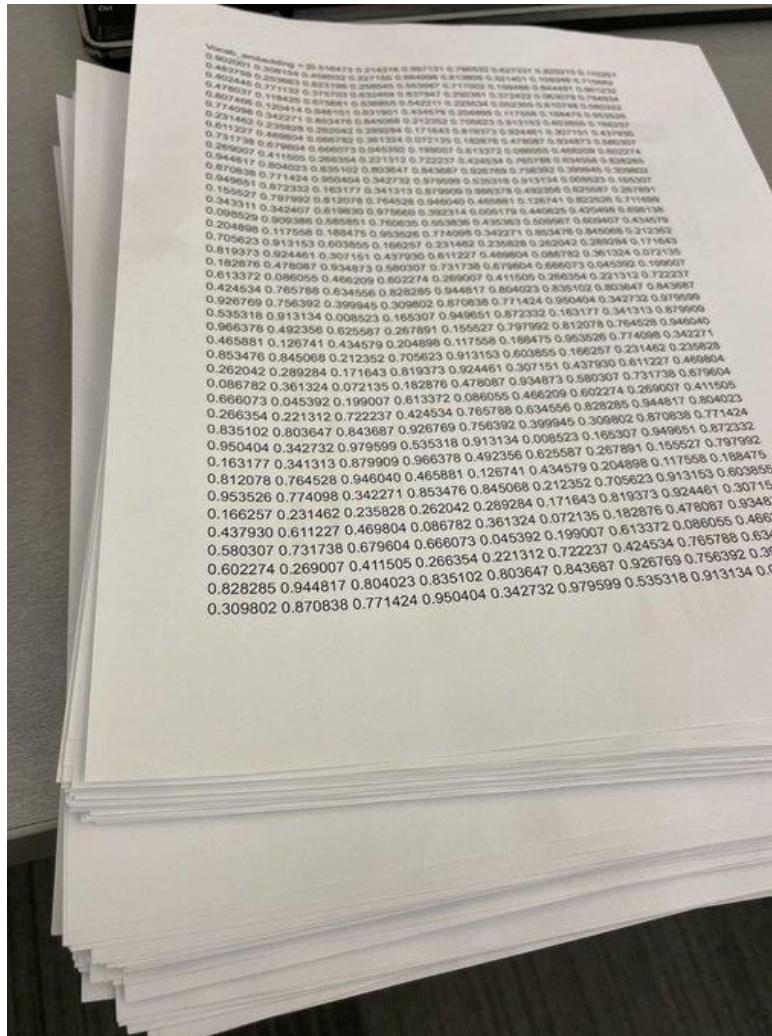
DS 530
Fairness and Interpretability in Data
Science

ENİS KAYIŞ

Breakout Groups

- ▶ Is ChatGPT an interpretable model?

ChatGPT Weights



175 billion weights pictured

Algorithm in Data Science

- ▶ Machines surpass humans in many tasks.
 - ▶ Advantages:
 - ▶ speed (faster than humans)
 - ▶ reproducibility (reliably delivers consistent results)
 - ▶ scaling (and can be copied infinitely)
 - ▶ Disadvantage:
 - ▶ insights about the data and the task the machine solves is hidden in increasingly complex models
 - millions of numbers to describe a deep neural network
 - best performing models are often ensembles that cannot be interpreted, even if each single model could be interpreted
 - ▶ Only focus is on performance → more and more opaque models (boosted trees or deep neural networks)

What is interpretability?

- ▶ Difficult to (mathematically) define interpretability.
- ▶ Some definitions of interpretability:
 - ▶ **Interpretability is the degree to which a human can understand the cause of a decision.**
 - ▶ **Interpretability is the degree to which a human can consistently predict the model's result.**
- ▶ A model is better interpretable than another model if its outputs are easier for a human to comprehend than outputs from the other model.

The need for interpretability

- ▶ If a model performs well, **why do we not just trust the model** and ignore **why** it made a certain decision?
- ▶ A single metric, such as classification accuracy, is an incomplete description of most real-world tasks.
- ▶ What would you like to know after predictive modelling?
 1. **What** is predicted?
 2. **Why** was the prediction made?
- ▶ Would you pay for the interpretability with a drop in predictive performance?
 - ▶ Depends!!!

The need for interpretability

- ▶ Incompleteness in problem formalization:
 - ▶ Not enough to get the prediction (the **what**).
 - ▶ Must also explain how it came to the prediction (the **why**)
- ▶ **Human Learning Process**
 - ▶ Mental update when something unexpected happens
 - ▶ Unexpected events makes us curious
 - ▶ Why did this happen? An explanation
- ▶ To satisfy curiosity as to why certain predictions are created, interpretability is crucial.
- ▶ The more a model's output affects a person's life, the more important it is to explain.
 - ▶ If a loan application is rejected by the model, this may be completely unexpected for the applicants. They can only reconcile this inconsistency between expectation and reality with some kind of explanation.
 - ▶ The explanations do not actually have to fully explain the situation but should address a main cause.
- ▶ Any experience with model explanations provided to the users?

The need for interpretability

- ▶ **Goal of science:** gain knowledge
 - ▶ Many problems are solved with black box ML models
 - ▶ The model itself becomes the source of knowledge.
 - ▶ Interpretability makes it possible to extract this additional knowledge captured by the model.
- ▶ **Safety:** Error-free solutions
 - ▶ How to think about edge cases following the ML model prediction?
- ▶ **Detecting Bias:**
 - ▶ Models easily pick up biases from the data, which turns them into racists
 - ▶ Example: Loan Applications
 - ▶ What is the main objective?
 - grant loans only to people who will eventually repay them?
 - + not to discriminate on the basis of certain demographics
 - ▶ Interpretability is a useful debugging tool for **detecting bias** in machine learning models.

The need for interpretability

► **Social acceptance**

- We are integrating algorithms into our daily lives
- A machine or algorithm that explains its predictions will find more acceptance.
- Machines have to “**persuade**” us, so that they can achieve their intended goal.

► **Accountability**

- Machine learning models can only be **debugged and audited** when they can be interpreted

Interpretability

- ▶ It is easier to check the following properties for an interpretable model:
 - ▶ Fairness
 - ▶ Privacy
 - ▶ Reliability
 - ▶ Causality
 - ▶ Trust

Always look for interpretability?

► **When we do not need interpretability:**

- ▶ the model **has no significant impact**
- ▶ the **problem is well studied**
- ▶ people or programs have high incentives and means
to manipulate the system

Taxonomy of Interpretability Methods

- ▶ **Modelling Stage:**
 - ▶ **Intrinsic:** restricting the complexity of the model
 - ▶ E.g.: Sparse Linear models, Decision Trees
 - ▶ **Post hoc:** analyzing the model after training
 - ▶ E.g.: Permutation feature importance
- ▶ **Results / Outputs:**
 - ▶ **Feature summary statistic**
 - ▶ Summary statistics for each feature or pairs
 - ▶ **Feature summary visualization**
 - ▶ Table vs plot (partial dependence)
 - ▶ **Model internals:**
 - ▶ E.g.: learned weights
 - ▶ **Data point:**
 - ▶ Counterfactual explanations: Finding a similar data point by changing some of the features for which the predicted outcome changes
 - ▶ **Intrinsically interpretable model:**
 - ▶ Approximate the black-box model via an interpretable model

Taxonomy of Interpretability Methods

► Modelling Type:

- ▶ **Model-specific:** limited to specific model classes
 - ▶ E.g.: interpretation of regression weights
- ▶ **Model-agnostic :** used on any model and are applied after the model has been trained (post-hoc)
 - ▶ Don't have access to model internals such as weights or structural information

► Local vs Global:

- ▶ **Local:** explain an individual prediction
- ▶ **Global:** explain the entire model

Scope of Interpretability

- ▶ **Algorithm Transparency**
 - ▶ How does the algorithm create the model?
- ▶ **Global, Holistic Model Interpretability**
 - ▶ How does the trained model make predictions?
- ▶ **Global Model Interpretability on a Modular Level**
 - ▶ How do parts of the model affect predictions?
- ▶ **Local Interpretability for a Single Prediction**
 - ▶ Why did the model make a certain prediction for an instance?
- ▶ **Local Interpretability for a Group of Predictions**
 - ▶ Why did the model make specific predictions for a group of instances?

Evaluation of Interpretability

- ▶ Let's agree on something: No real consensus about what interpretability is and how to measure it
- ▶ Application-level evaluation
 - ▶ Test the explanations with domain experts (end users) within the product
- ▶ Human level evaluation
 - ▶ Test with laypersons (cheaper), let them choose among different explanations
- ▶ Function level evaluation
 - ▶ No humans required
 - ▶ E.g.: depth of a decision tree

Properties of Explanation Methods

- ▶ Expressive Power:
 - ▶ the “language” or structure of the explanations
 - ▶ E.g.: If-then rules, decision trees, linear regression
- ▶ Translucency: (Black or White Box)
 - ▶ How much the explanation method relies on looking into the model?
 - ▶ Zero translucency: Relying on manipulating inputs and observing the predictions
 - ▶ Highly translucent: Relying on intrinsically interpretable models
- ▶ Portability:
 - ▶ The range of machine learning models with which the explanation method can be used
 - ▶ Low translucency implies higher portability
- ▶ Algorithmic Complexity
 - ▶ Computational time is a constraint

} Black box is better
for portability

Human-friendly Explanations

- ▶ What do you look for in an explanation?
- ▶ Humans prefer short explanations (only 1 or 2 causes)
 - ▶ contrast the current situation with a situation in which the event would not have occurred
 - ▶ do not want a complete explanation for a prediction
- ▶ Abnormal causes provide good explanations.
- ▶ Explanations are social interactions between the explainer and the explainee

Interpretable Models

- ▶ Linear Regression
- ▶ Logistic Regression
- ▶ GLM, GAM and more
- ▶ Decision Tree
- ▶ Decision Rules
- ▶ RuleFit
- ▶ Others (Naïve Bayes, KNN)

Linear Regression

- ▶ Predicts the target as a weighted sum of the feature inputs:

$$y = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p + \epsilon$$

- ▶ Betas represent the learned feature weights
- ▶ The first weight in the sum (β_0) is called the intercept and is not multiplied with a feature.
- ▶ The epsilon (ϵ) is the error: difference between the prediction and the actual outcome, assumed to be normally distributed (many small and few large errors)
- ▶ Easy interpretation

Linear Regression

- ▶ Estimation of the weights:
 - ▶ OLS is the most common one

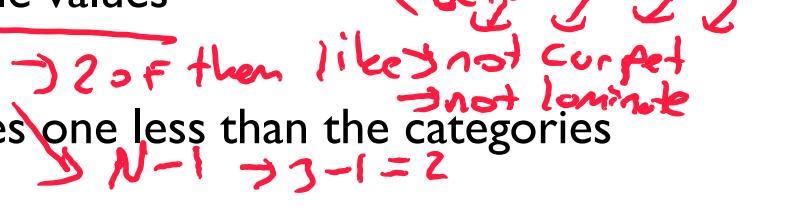
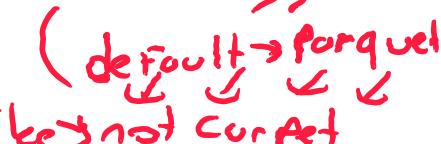
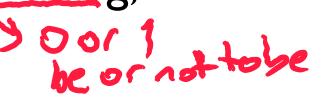
$$\hat{\beta} = \arg \min_{\beta_0, \dots, \beta_p} \sum_{i=1}^n \left(y^{(i)} - \left(\beta_0 + \sum_{j=1}^p \beta_j x_j^{(i)} \right) \right)^2$$

- ▶ Many other methods are also available (e.g., LAD, robust)
- ▶ Estimated weights come with confidence intervals:
 - ▶ What does a 95% confidence interval for, say β_2 , ranges from 1 to 3 mean?
- ▶ Linear regression models are **linear**:
 - ▶ Estimation is (generally) easy
 - ▶ Interpretation is easy
 - ▶ Ex: We may need to (1) predict the clinical outcome of a patient and (2) quantify the influence of the drug and while taking sex, age, etc. into account in an interpretable way.

Linear Regression

- ▶ BE AWARE of the underlying assumptions:
 - ▶ Linearity of the model
 - ▶ greatest strength **BUT** greatest limitation
 - ▶ Normality of the errors (and the target outcome)
 - ▶ If violated, confidence intervals are not correct
 - ▶ Homoscedasticity (constant variance) over the entire predictor space
 - ▶ *Each value has some Variance between others.*
 - ▶ Independence of instances
 - ▶ If violated (multiple measurements), use mixed effect models
 - ▶ Fixed Features
 - ▶ No measurement errors in the features
 - ▶ No multicollinearity
 - ▶ If violated, the estimation of the weights are messed up

Linear Regression

- ▶ Interpretations:
 - ▶ Numerical features
 - ▶ Increasing it by one unit changes the estimated outcome by its weight
 - ▶ Binary features
 - ▶ Ex: "House has a garage or not" 
 - ▶ Changing it from the reference category to the other category changes the estimated outcome by the feature's weight
 - ▶ Categorical feature with multiple categories
 - ▶ A feature with a fixed number of possible values
 - ▶ Ex: Floor Type: carpet, laminate, parquet 
 - ▶ One-hot-encoding, need dummy variables one less than the categories 
 - ▶ Intercept 
 - ▶ For an instance with all numerical feature values at zero and the categorical feature values at the reference categories, the model prediction is the intercept weight
 - ▶ All the interpretations always come with the footnote that "all other features remain the same".

Linear Regression

► Model Fit

► R-Squared

- how much of the total variance of your target outcome is explained by the model

lower better

Sum of Squared Error

$$R^2 = \underbrace{1 - \frac{SSE}{SST}}_{\text{Higher better}}$$

How well our chosen independent variable (x) explains the variance on dependent variable (y)

$$SSE = \sum_{i=1}^n (y^{(i)} - \hat{y}^{(i)})^2$$

$$SST = \sum_{i=1}^n (y^{(i)} - \bar{y})^2$$

Sum of Squares Total

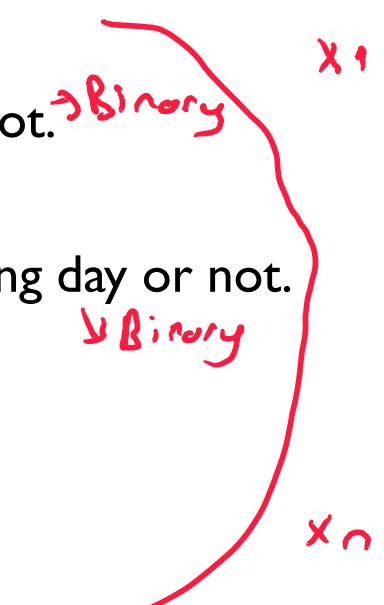
- Generally ranges from 0 to 1 (negative values are possible?)
- increases with the number of features in the model
- Adjusted R-squared

$$R^2 = 1 - (1 - R^2) \frac{n - 1}{n - p - 1}$$

Linear Regression

▶ Example:

▶ Bike Rentals

- ▶ Target Variable: Count of bicycles → y
 - ▶ Season: spring, summer, fall or winter. → Categorical
 - ▶ Holiday indicator: whether the day was a holiday or not. → Binary
 - ▶ The year: 2011 or 2012 → Numerical
 - ▶ Working Day indicator :whether the day was a working day or not.
 - ▶ The weather situation → Categorical
 - ▶ Temperature → Numerical
 - ▶ Relative humidity in percent → Numerical
 - ▶ Wind speed (km per hour) → Numerical
- 

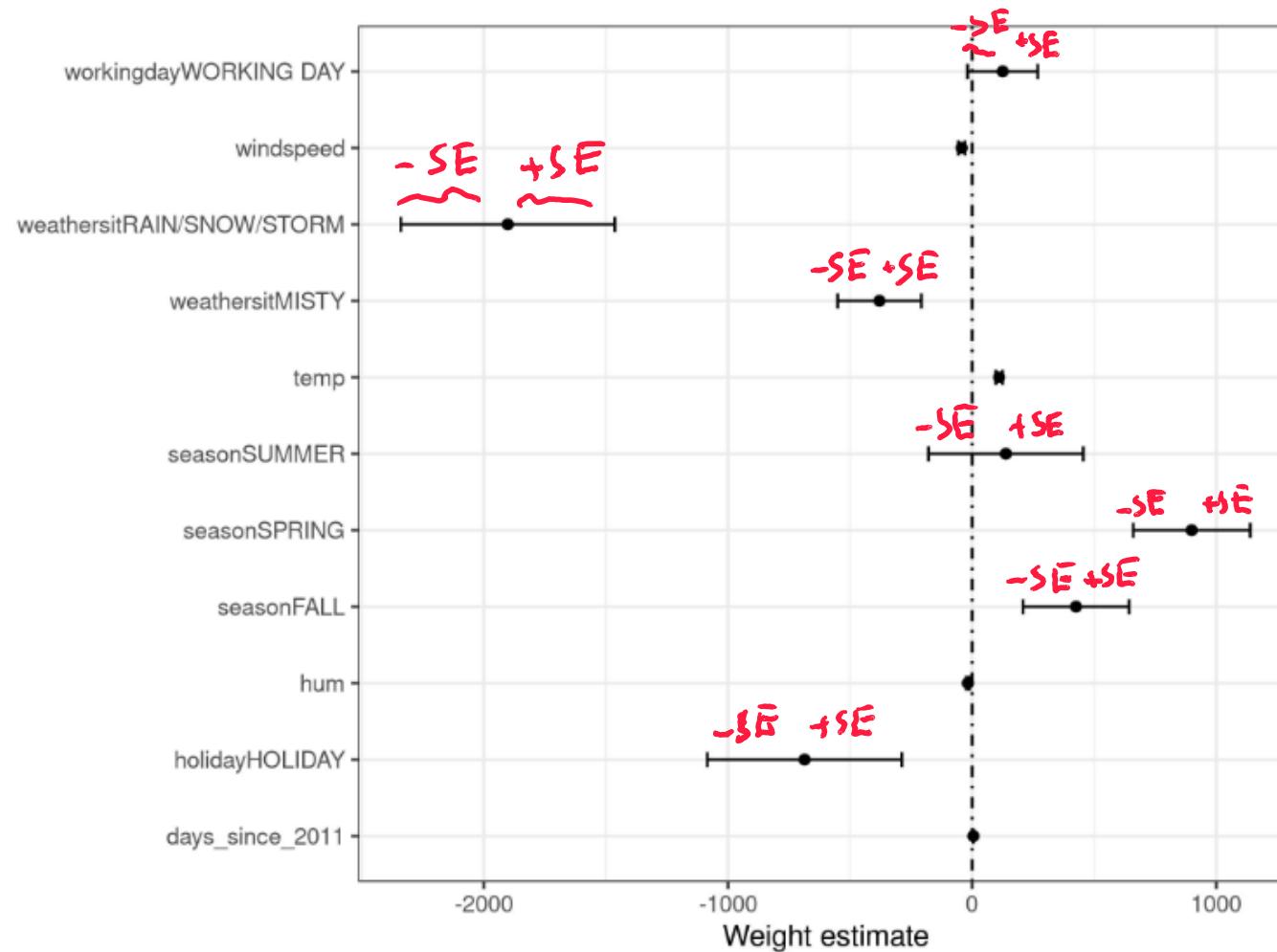
Linear Regression

- ▶ Example:
- ▶ Bike Rentals

	Weight	SE	t
(Intercept) <i>(season Winter + Not Holiday not workingday)</i>	2399.4	238.3	10.1
seasonSPRING	899.3	122.3	7.4
seasonSUMMER	138.2	161.7	0.9
seasonFALL	425.6	110.8	3.8
holidayHOLIDAY	-686.1	203.3	3.4
workingdayWORKING DAY	124.9	73.3	1.7
weathersitMISTY	-379.4	87.6	4.3
weathersitRAIN/SNOW/STORM	-1901.5	223.6	8.5
temp	110.7	7.0	15.7
hum	-17.4	3.2	5.5
windspeed	-42.5	6.9	6.2
days_since_2011	4.9	0.2	28.5

Linear Regression

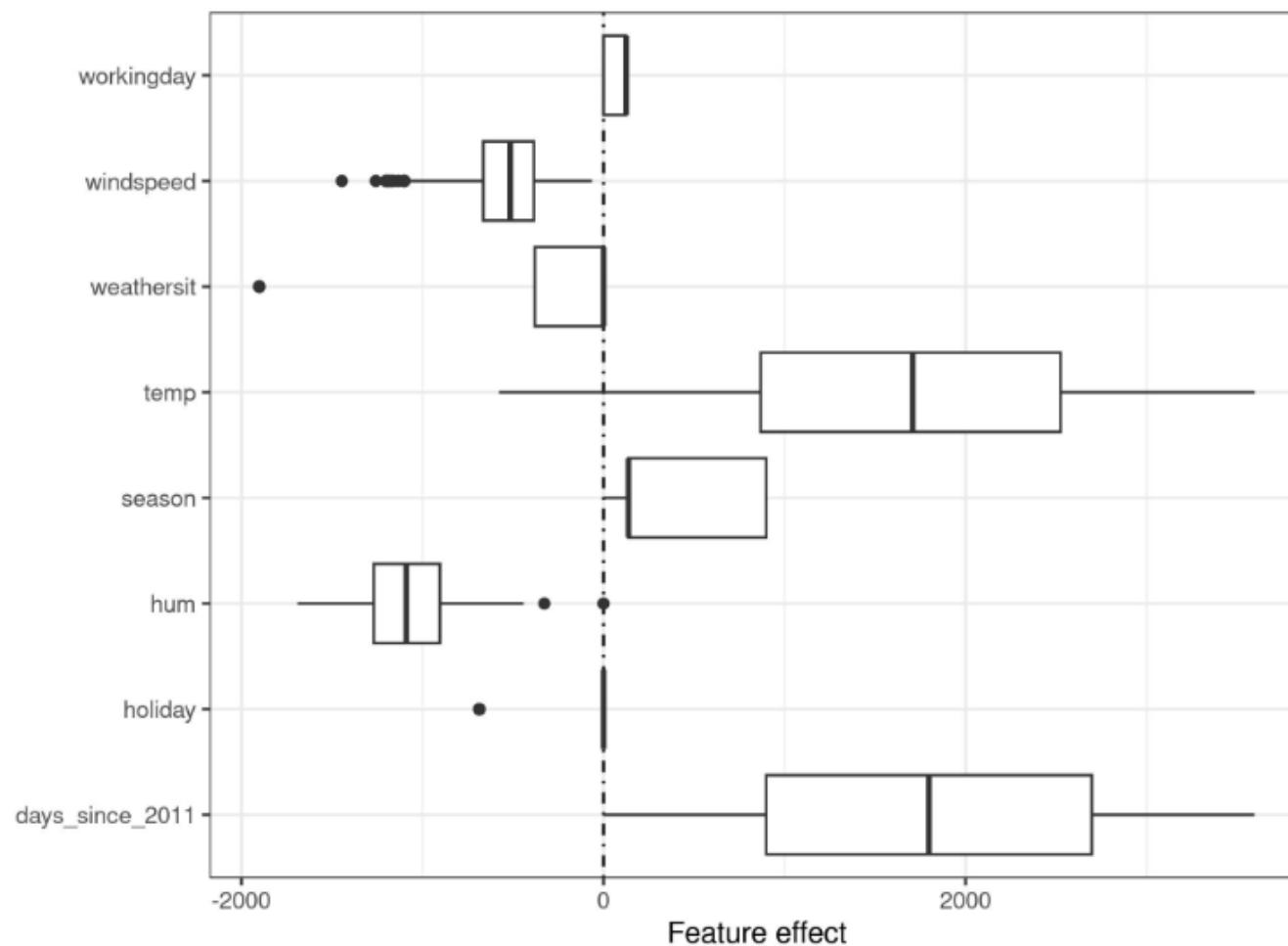
- ▶ Example: Bike Rentals
- ▶ Weight Plot



Linear Regression

▶ Example: Bike Rentals

▶ Effect Plot $\text{effect}_j^{(i)} = w_j x_j^{(i)}$



Linear Regression

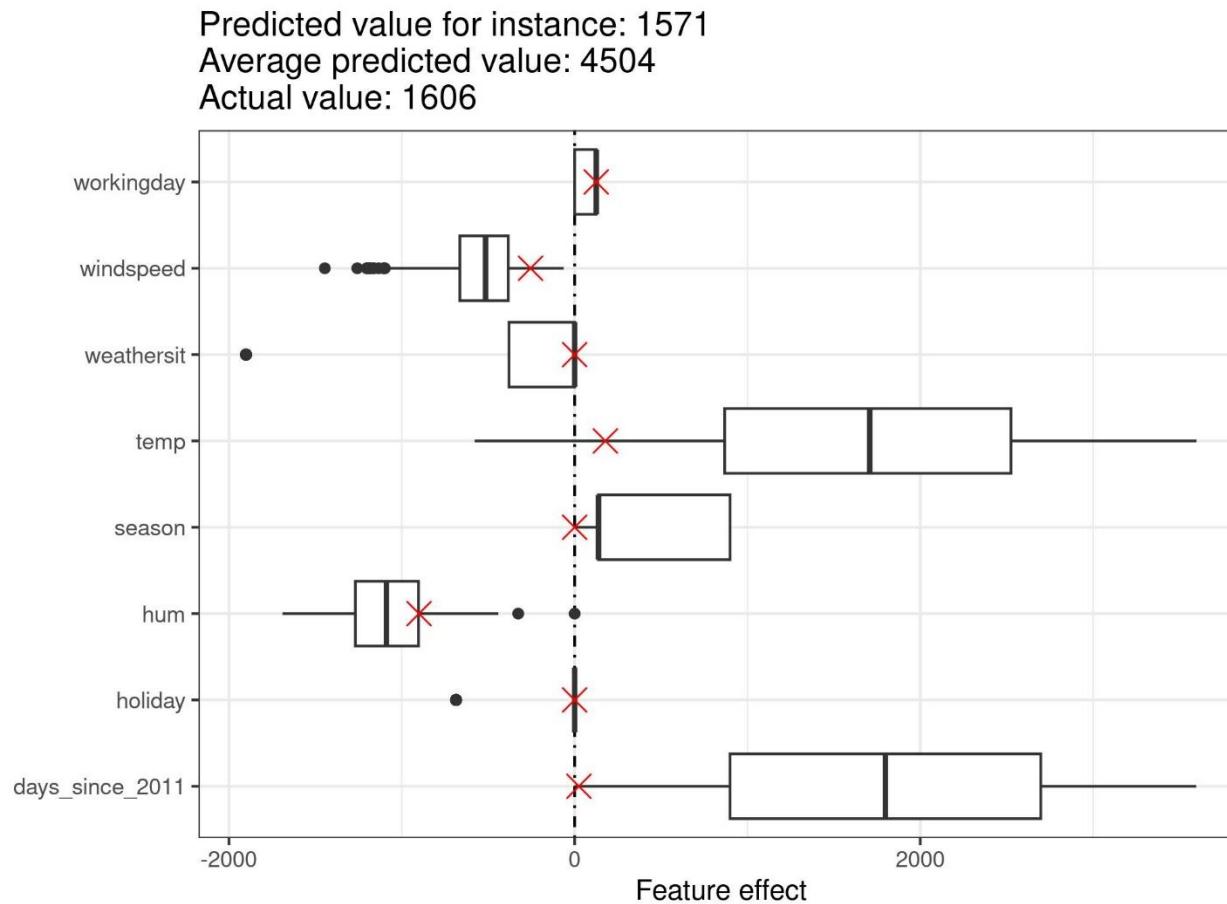
- ▶ Example: Bike Rentals
 - ▶ Explain Individual Predictions (feature values for instance #6)

Feature	Value
season	WINTER
yr	2011
mnth	JAN
holiday	NO HOLIDAY
weekday	THU
workingday	WORKING DAY
weathersit	GOOD
temp	1.604356
hum	51.8261
windspeed	6.000868
cnt	1606
days_since_2011	5

Linear Regression

▶ Example: Bike Rentals

- ▶ Explain Individual Predictions (ex: feature values for instance #6)



Linear Regression

- ▶ Do Linear Models Create Good Explanations?
 - ▶ Not necessarily
- ▶ **Contrastive?** Reference instance is where all numerical features are zero and the categorical features are at their reference categories.
- ▶ Do you think this is likely to occur?
- ▶ Linear models do not create selective explanations by default.
- ▶ Using less features or by training sparse linear models.
- ▶ Linear models create truthful explanations, as long as the linear equation is appropriate between features and outcome.
- ▶ Linearity makes the explanations more general and simpler.
 - “The house is expensive because it is big”

Linear Regression

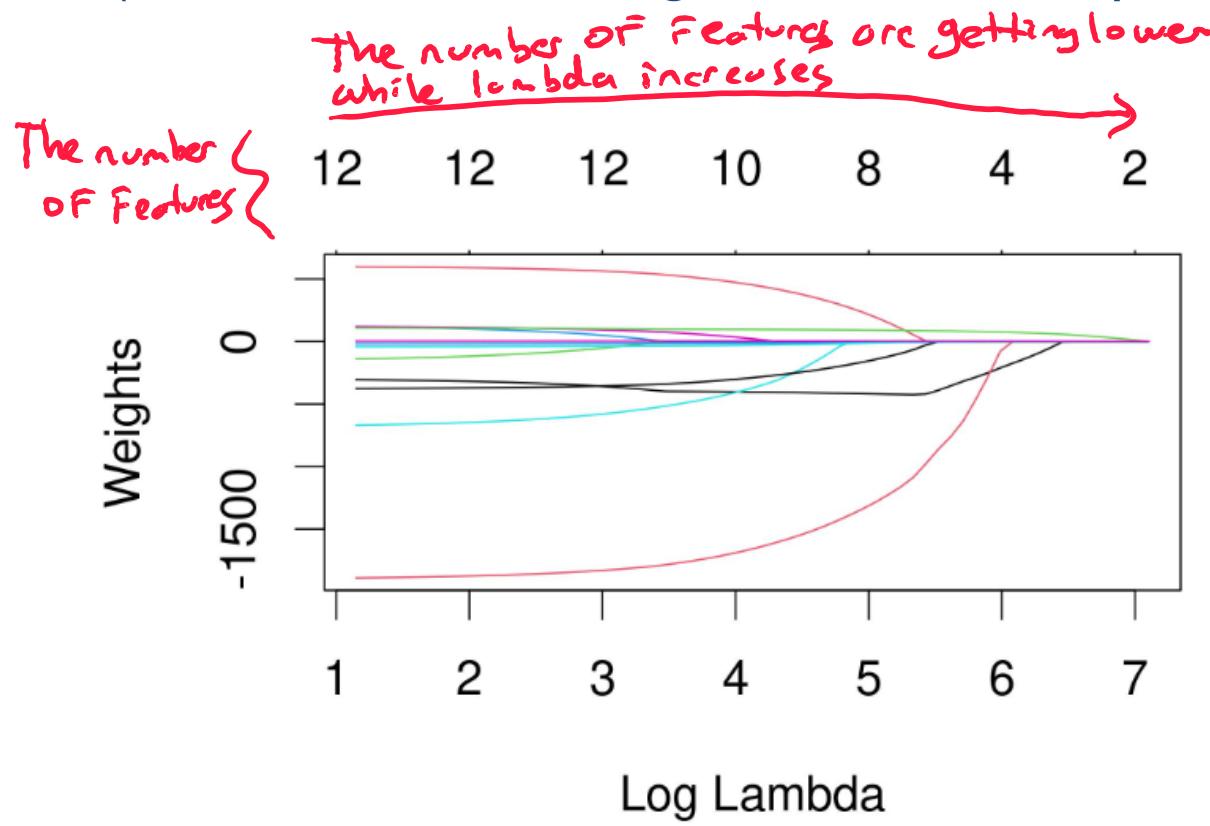
- ▶ What happens to interpretability if you have hundreds of features?
- ▶ Sparse Linear Models
 - ▶ Lasso (least absolute shrinkage and selection operator):
 - ▶ An automatic way to introduce sparsity → apart from each other Scatterness,
 - ▶ Performs feature selection and regularization of the feature weights,

$$\min_{\beta} \left(\frac{1}{n} \sum_{i=1}^n (y^{(i)} - x_i^T \beta)^2 + \lambda \|\beta\|_1 \right)$$

- ▶ Many of the weights receive an estimate of 0 and the others are shrunk.
- ▶ Lambda controls the strength of the regularizing effect and is tuned by cross-validation. (kinda most important)
- The larger the lambda parameter, the fewer features are present in the model (because their weights are zero) and the better the model can be interpreted.

Linear Regression

- ▶ Sparse Linear Models
 - ▶ Lasso (least absolute shrinkage and selection operator):



Linear Regression

- ▶ Lasso (least absolute shrinkage and selection operator):
 - ▶ Ex: Bicycle Rentals
 - ▶ Caution: Are the features standardized?
 - ▶ If they are, make sure to transform the weights back to match the original feature scales.

The model chooses them even if they are really expensive bc they are most important

	High λ	Low λ
	Weight	Weight
seasonWINTER	0.00	-389.99
seasonSPRING	0.00	0.00
seasonSUMMER	0.00	0.00
seasonFALL	0.00	0.00
holidayHOLIDAY	0.00	0.00
workingdayWORKING DAY	0.00	0.00
weathersitMISTY	0.00	0.00
weathersitRAIN/SNOW/STORM	0.00	-862.27
temp	52.33	85.58
hum	0.00	-3.04
windspeed	0.00	0.00
days_since_2011	2.15	3.82

Linear Regression

- ▶ Sparse Linear Models
 - ▶ Manually selected features
 - ▶ Makes sense if you have access to an expert
 - ▶ No automation
 - ▶ Univariate selection
 - ▶ Select features that are highly correlated (pairwise) with the target outcome.
 - ▶ WARNING: Some features might not show a correlation until some other features are accounted for.
 - ▶ Stepwise methods
 - ▶ Forward selection
 - Start with one feature with the highest R², then gradually add new ones
 - ▶ Backward selection
 - Start with all the features, then gradually remove insignificant ones

Linear Regression

► Advantages:

- **Transparent:** the predictions as a **weighted sum**
- **Simple:** Control for the **number of features** in your model
- **High level** of **collective experience** and **expertise**
- **Guarantee** to find **optimal weights**

► Disadvantages:

- Each **nonlinearity** or **interaction** must be **hand-crafted**
- Often not that good regarding predictive performance
- The **interpretation** of a **weight** can be **unintuitive**, because it depends on all other features

→ Composed
to Neural
Networks

ÖZYEĞİN ÜNİVERSİTESİ

DS 530

Fairness and Interpretability

ENİS KAYIŞ

Interpretable Models

- ▶ Linear Regression
- ▶ Logistic Regression
- ▶ GLM, GAM and more
- ▶ Decision Tree
- ▶ Decision Rules
- ▶ RuleFit
- ▶ Others (Naïve Bayes, KNN)

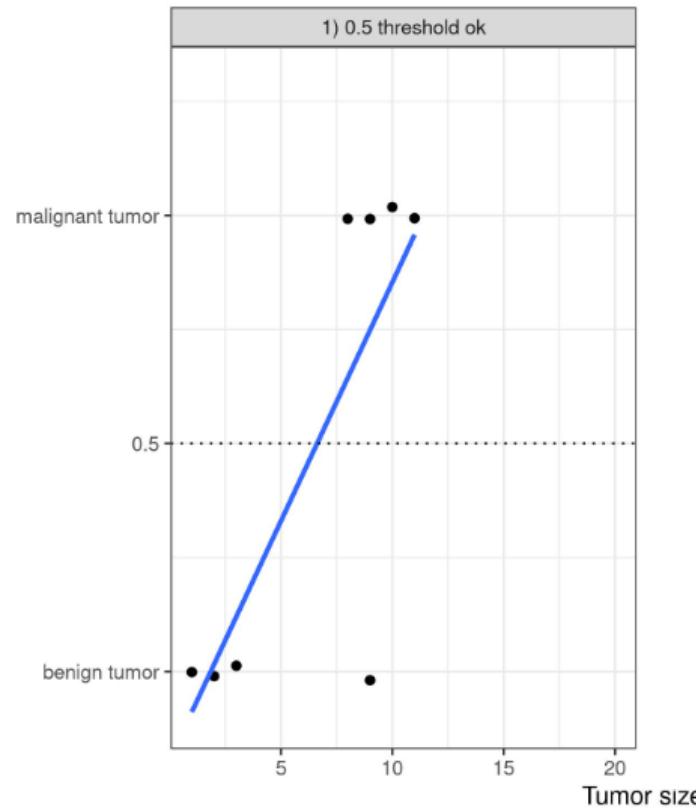
Logistic Regression

- ▶ Logistic regression models the probabilities for classification problems with two possible outcomes.
- ▶ The linear regression model can work well for regression (when y is numerical) but fails for classification (when y is categorical). Why?
 - ▶ Technically no issues, treat y classes as a numerical feature
 - ▶ A linear model does not output probabilities, just a linear interpolation between points
 - ▶ Gives you values below zero and above one
 - ▶ No meaningful threshold at which you can distinguish one class from the other

Logistic Regression

▶ Consider this example:

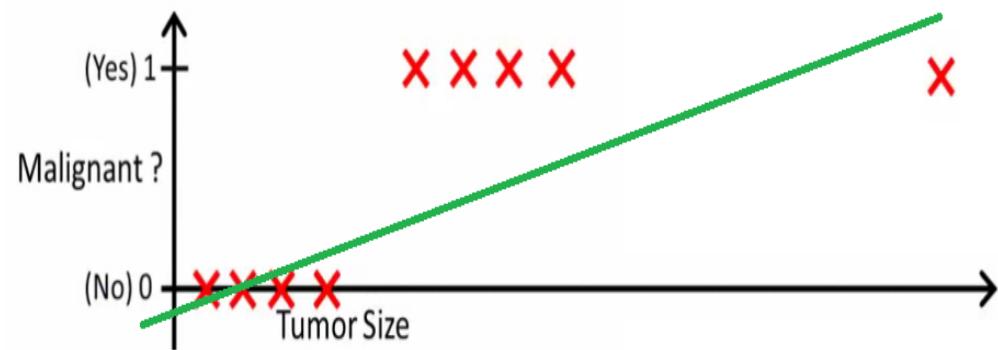
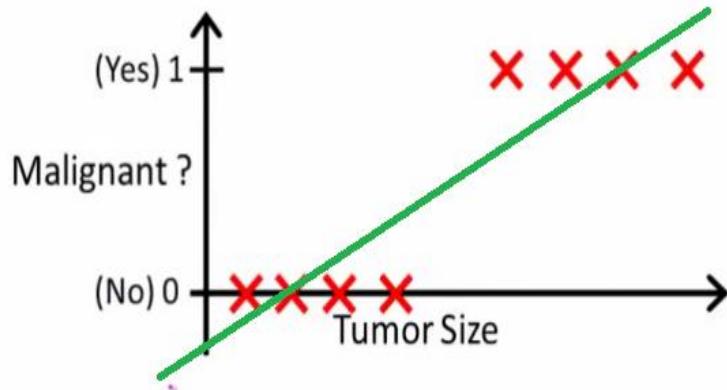
- ▶ Malignant tumors get 1 and non-malignant ones get 0
- ▶ Fit a straight line through {tumor size, tumor type} (green line, call $h(x)$)
- ▶ for any given tumor size x , if $h(x) > 0.5$ predict malignant tumor, otherwise predict benign.



Logistic Regression

▶ Consider this example:

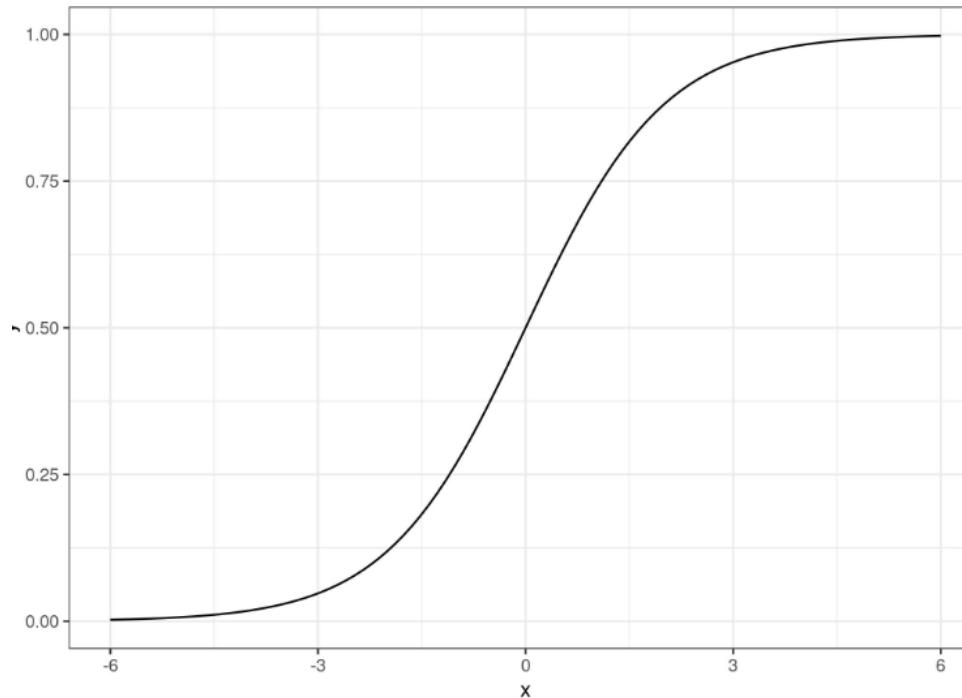
- ▶ Let's add a few more data points to the original sample (left panel) and run linear regression again (right panel)
- ▶ Need a new threshold for $h(x)$, maybe 0.6?
- ▶ We cannot change the threshold each time a new sample arrives!!!



Logistic Regression

- ▶ A solution: Logistic regression
 - ▶ Uses the logistic function to squeeze the output of a linear equation between 0 and 1.

$$\text{logistic}(\eta) = \frac{1}{1 + \exp(-\eta)}$$



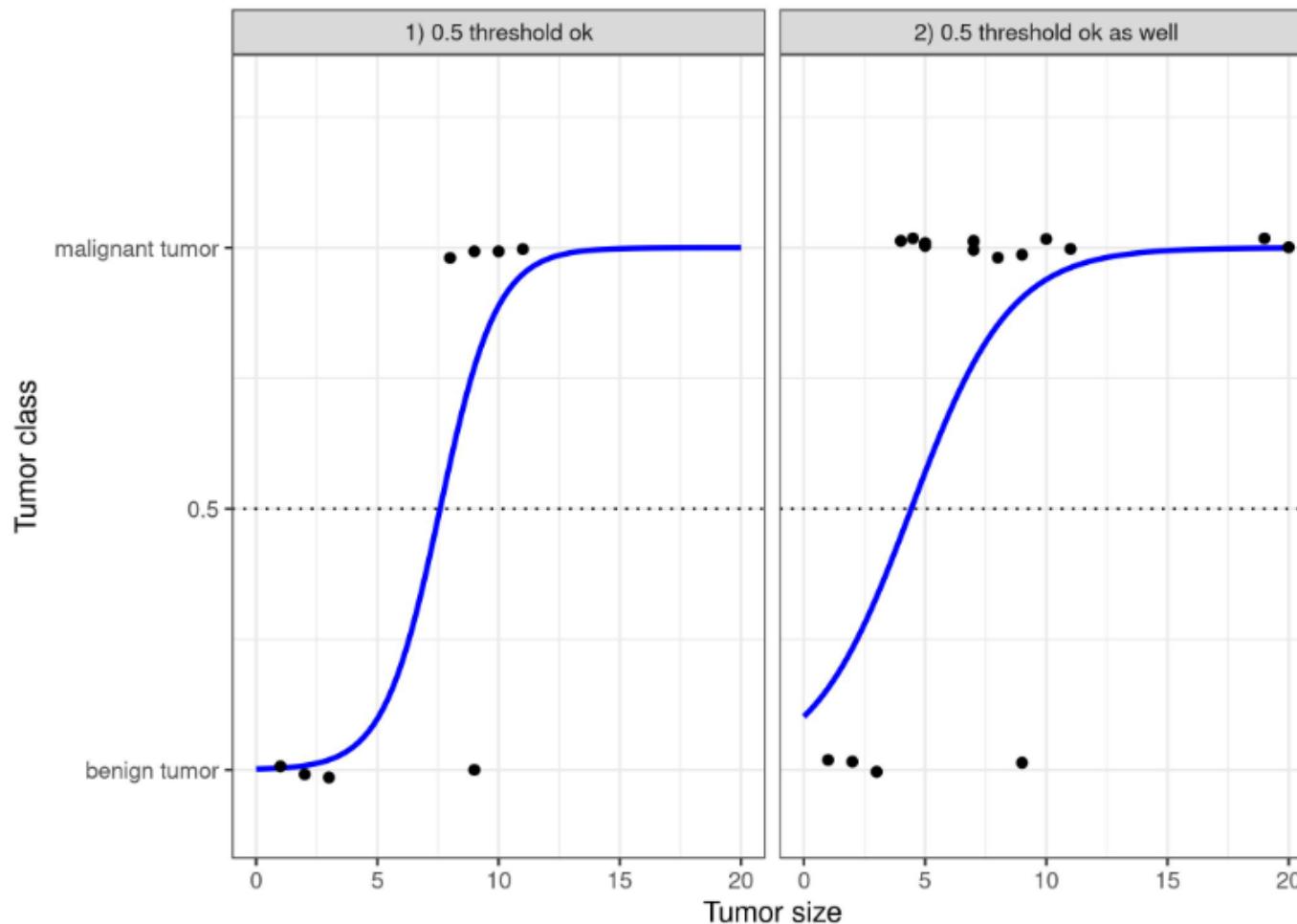
Logistic Regression

- We desire probabilities between 0 and 1:

$$P(y^{(i)} = 1) = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x_1^{(i)} + \dots + \beta_p x_p^{(i)}))}$$

Logistic Regression

- ▶ Let's revisit the previous example
 - ▶ we can use 0.5 as a threshold in both cases



Logistic Regression

▶ Interpretation

- ▶ Interpretation of weights in logistic regression differs from linear regression: the outcome is a probability
- ▶ The weights do not influence the probability linearly
- ▶ Weighted sum is transformed by the logistic function to a probability
- ▶ So, what does the linear term predict?

$$\ln \left(\frac{P(y=1)}{1 - P(y=1)} \right) = \log \left(\frac{P(y=1)}{P(y=0)} \right) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

- ▶ Log odds
 - ▶ Odds: probability of event divided by probability of no event
- ▶ Logistic regression is a linear model for the log odds. Great! ☺

Logistic Regression

- ▶ To see the effect of each feature, rewrite to get

$$\frac{P(y = 1)}{1 - P(y = 1)} = \text{odds} = \exp(\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p)$$

- ▶ Now, let's increase x_j by 1 unit

$$\begin{aligned}\frac{\text{odds}_{x_j+1}}{\text{odds}_x_j} &= \frac{\exp(\beta_0 + \beta_1 x_1 + \dots + \beta_j(x_j + 1) + \dots + \beta_p x_p)}{\exp(\beta_0 + \beta_1 x_1 + \dots + \beta_j x_j + \dots + \beta_p x_p)} \\ &= \exp(\beta_j(x_j + 1) - \beta_j x_j) = \exp(\beta_j)\end{aligned}$$

- ▶ A change in a feature by one unit changes the odds ratio (multiplicative) by a factor of $\exp(\beta_j)$

Logistic Regression

- ▶ Interpretation:
 - ▶ A change in x_j by one unit increases the log odds ratio by β_j .
 - ▶ Having odds of 2: the probability for $y=1$ is twice as high as $y=0$.
 - ▶ If $\beta_j = 0.7$, then increasing the respective feature by one unit multiplies the odds by $\exp(0.7)$ (~2) and the odds change to 4 ($=2*2$).
 - ▶ To calculate the odds, one need values for all the features, thus we usually interpret the weights as odds ratios.

Logistic Regression

- ▶ Interpretation of weights:
 - ▶ Numerical feature: If you increase the value of feature x_j by one unit, the estimated odds change by a **factor of** $\exp(\beta_j)$
 - ▶ Binary feature: Changing the feature x_j from the reference category to the other category changes the estimated odds by a **factor of** $\exp(\beta_j)$
 - ▶ Categorical feature: Similar to binary feature, after one-hot encoding
 - ▶ Intercept β_0 : When all numerical features are zero and the categorical features are at the reference category, the estimated odds are $\exp(\beta_0)$.

Logistic Regression

- ▶ Example:
 - ▶ Dataset: Cervical Cancer (Classification)
 - ▶ Age in years
 - ▶ Number of sexual partners
 - ▶ First sexual intercourse (age in years)
 - ▶ Number of pregnancies
 - ▶ Smoking yes or no
 - ▶ Smoking (in years)
 - ▶ Hormonal contraceptives yes or no
 - ▶ Hormonal contraceptives (in years)
 - ▶ Intrauterine device yes or no (IUD)
 - ▶ Number of years with an intrauterine device (IUD)
 - ▶ Has patient ever had a sexually transmitted disease (STD) yes or no
 - ▶ Number of STD diagnoses
 - ▶ Time since first STD diagnosis
 - ▶ Time since last STD diagnosis
 - ▶ **The biopsy results “Healthy” or “Cancer”. Target outcome.**

Logistic Regression

▶ Example: Cervical Cancer Prediction

	Weight	Odds ratio	Std. Error
Intercept	-2.91	0.05	0.32
Hormonal contraceptives y/n	-0.12	0.89	0.30
Smokes y/n	0.26	1.30	0.37
Num. of pregnancies	0.04	1.04	0.10
Num. of diagnosed STDs	0.82	2.27	0.33
Intrauterine device y/n	0.62	1.86	0.40

- ▶ An increase in the **number of diagnosed STDs** increases the odds of cancer by a factor of 2.27, when all other features remain the same.

Logistic Regression

▶ Example: Cervical Cancer Prediction

	Weight	Odds ratio	Std. Error
Intercept	-2.91	0.05	0.32
Hormonal contraceptives y/n	-0.12	0.89	0.30
Smokes y/n	0.26	1.30	0.37
Num. of pregnancies	0.04	1.04	0.10
Num. of diagnosed STDs	0.82	2.27	0.33
Intrauterine device y/n	0.62	1.86	0.40

- ▶ For women using hormonal contraceptives, the odds for cancer are **by a factor of 0.89 lower**, compared to women without hormonal contraceptives, given all other features stay the same.

Logistic Regression

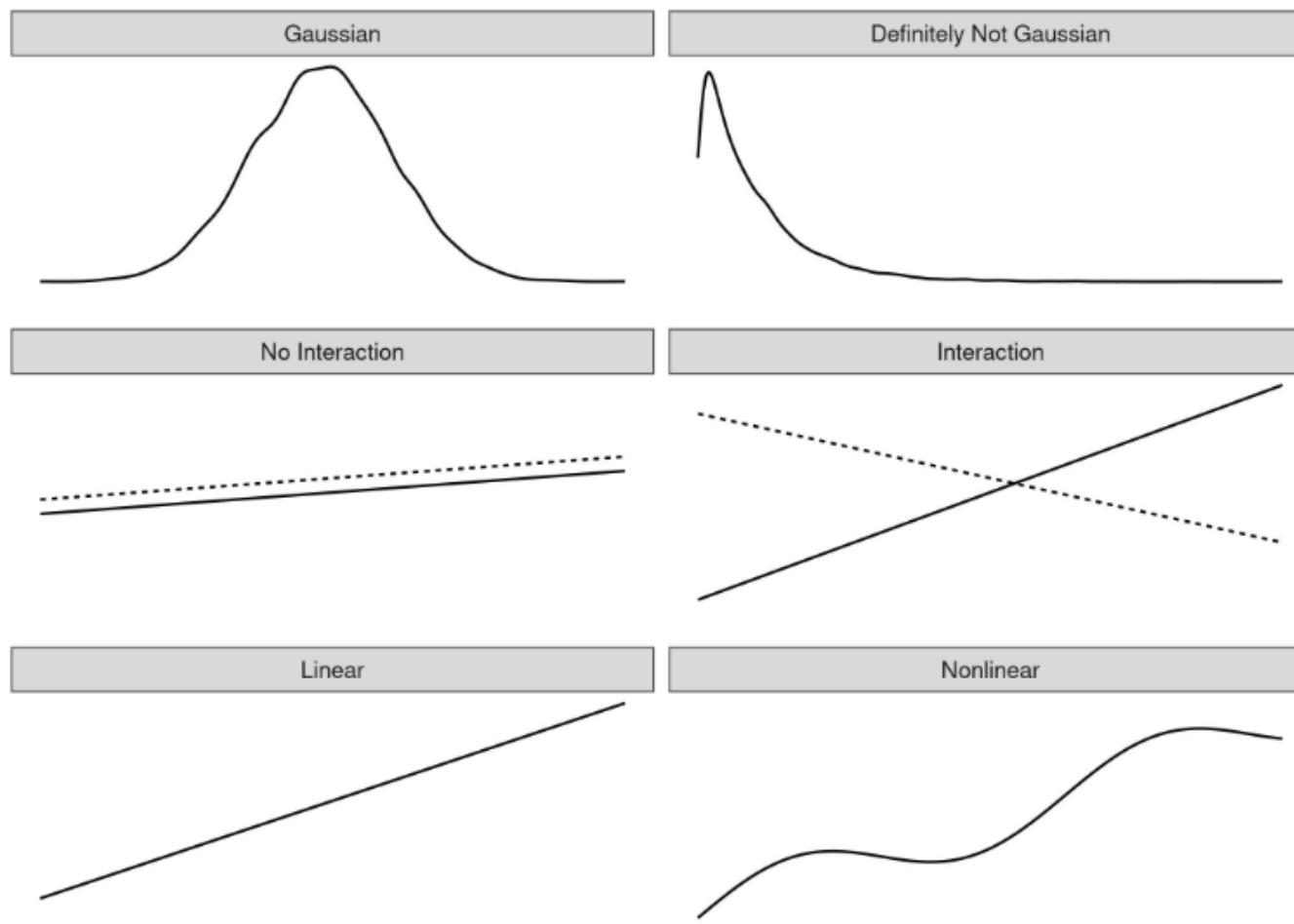
- ▶ Similar pros and cons with the linear regression model.
- ▶ Interpretation of the weights is even more difficult (effect of weights are multiplicative, not additive)
- ▶ Logistic regression can suffer from complete separation.
 - ▶ There is a feature that would perfectly separate the two classes
 - ▶ The weight for that feature would not converge, because the optimal weight would be infinite.
 - ▶ Is this really a problem?
- ▶ Logistic regression model is not only a classification model, but also gives you probabilities.
 - ▶ Knowing that an instance has a 99% probability for a class compared to 51% makes a big difference.

GLM and GAM

- ▶ Many assumptions behind linear regression
 - ▶ Bad news: They are often violated in reality
 - ▶ The outcome may have a non-Gaussian distribution
 - ▶ The features might interact
 - ▶ There could be a nonlinear relationship between the features and the outcome
 - ▶ Good News: Many variants of linear regression exist to remedy these problems
- ▶ Generalized Linear Models (GLMs)
- ▶ Generalized Additive Models (GAMs)

GLM and GAM

- ▶ Focus on three common violations of linear regression assumptions (there are more types of violations!!!)



Non-Gaussian Outcomes: GLM

- ▶ Assumption: The outcome given the input features follows a Normal distribution. Violated when the outcome is:
 - ▶ a category (returned vs not returned)
 - ▶ a count (number of children)
 - ▶ the time to the occurrence of an event (time to failure of a machine)
 - ▶ A very skewed outcome with a few very high values (household income).
- ▶ Generalized Linear Models (GLMs)
 - ▶ Handle all these type of outcomes

GLM

- ▶ Idea:
 - ▶ Keep the weighted sum of the features
 - ▶ BUT allow non-Normal outcome distributions
 - ▶ Connect the expected mean of this distribution and the weighted sum through a possibly nonlinear function.
 - ▶ Ex: Logistic regression model assumes a Bernoulli distribution for the outcome and links the expected mean and the weighted sum using the logistic function.
- ▶ Mathematically:

$$g(E_Y(y|x)) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

Link function

GLM

$$g(E_Y(y|x)) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

- ▶ GLMs consist of three components:
 - ▶ The link function $g()$,
 - ▶ The weighted sum $X^T \beta$
 - ▶ A probability distribution from the exponential family that defines E_Y .
- ▶ Example: Linear Regression
 - ▶ The link function: $g(x) = x$
 - ▶ Probability distribution: Normal
- ▶ Example: Logistic Regression
 - ▶ The link function: $g(x)$ is the logistic function
 - ▶ Probability Distribution: Bernoulli

GLM

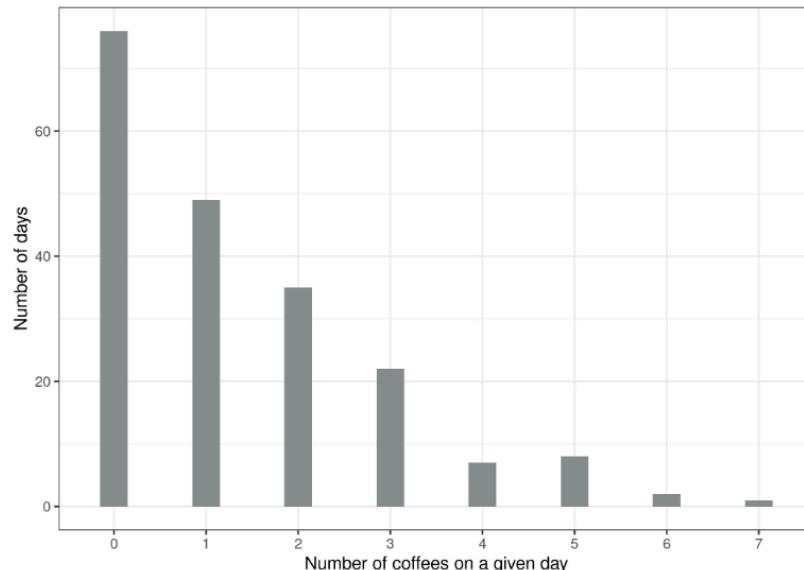
- ▶ Example: Number of coffees sold
 - ▶ The link function: $g(x)$ is the natural logarithm
 - ▶ Probability Distribution: Poisson

$$\ln(E_Y(y|x)) = x^T \beta$$

- ▶ How to choose the right link function?
- ▶ Consider:
 - ▶ the distribution of your target
 - ▶ how well the model fits your actual data

GLM

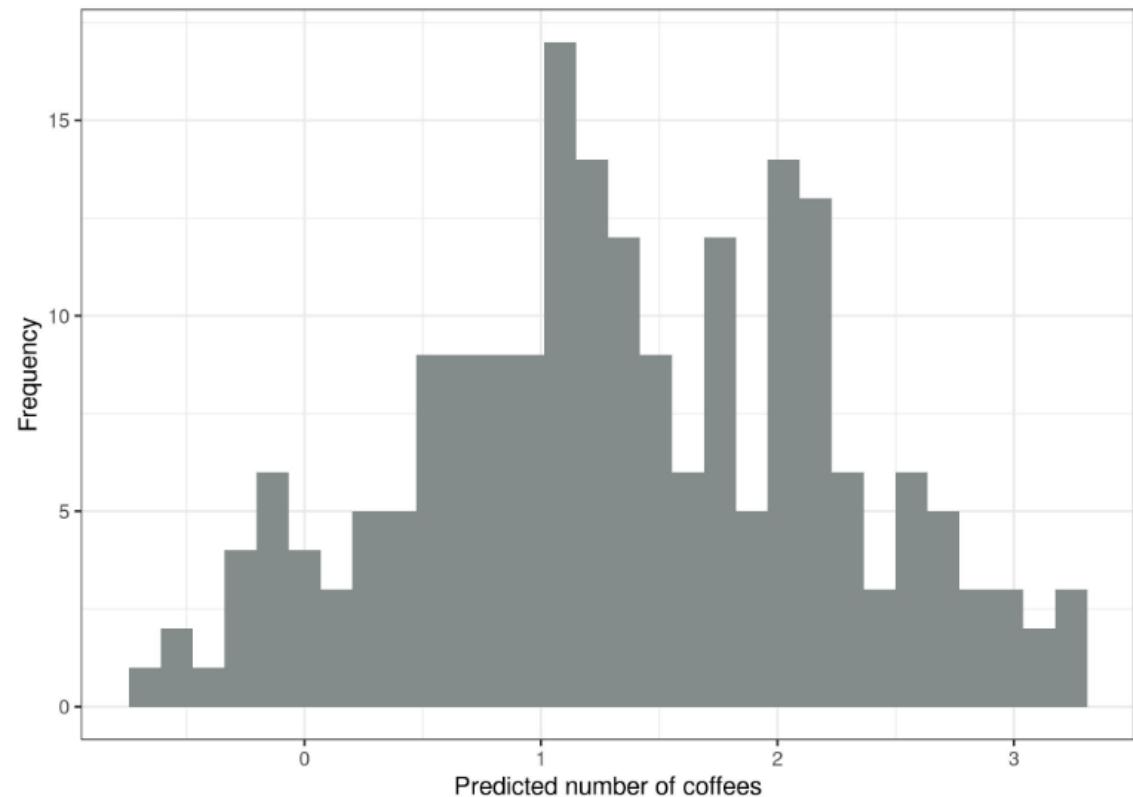
- ▶ Example: Number of coffees drank
 - ▶ You have collected data about your daily coffee drinking behavior.
 - ▶ The number of cups (target variable)
 - ▶ Your stress level on a scale of 1 to 10
 - ▶ How well you slept the night before on a scale of 1 to 10
 - ▶ Whether you had to work on that day.



On 76 of the 200 days you had no coffee at all and on the most extreme day you had 7.

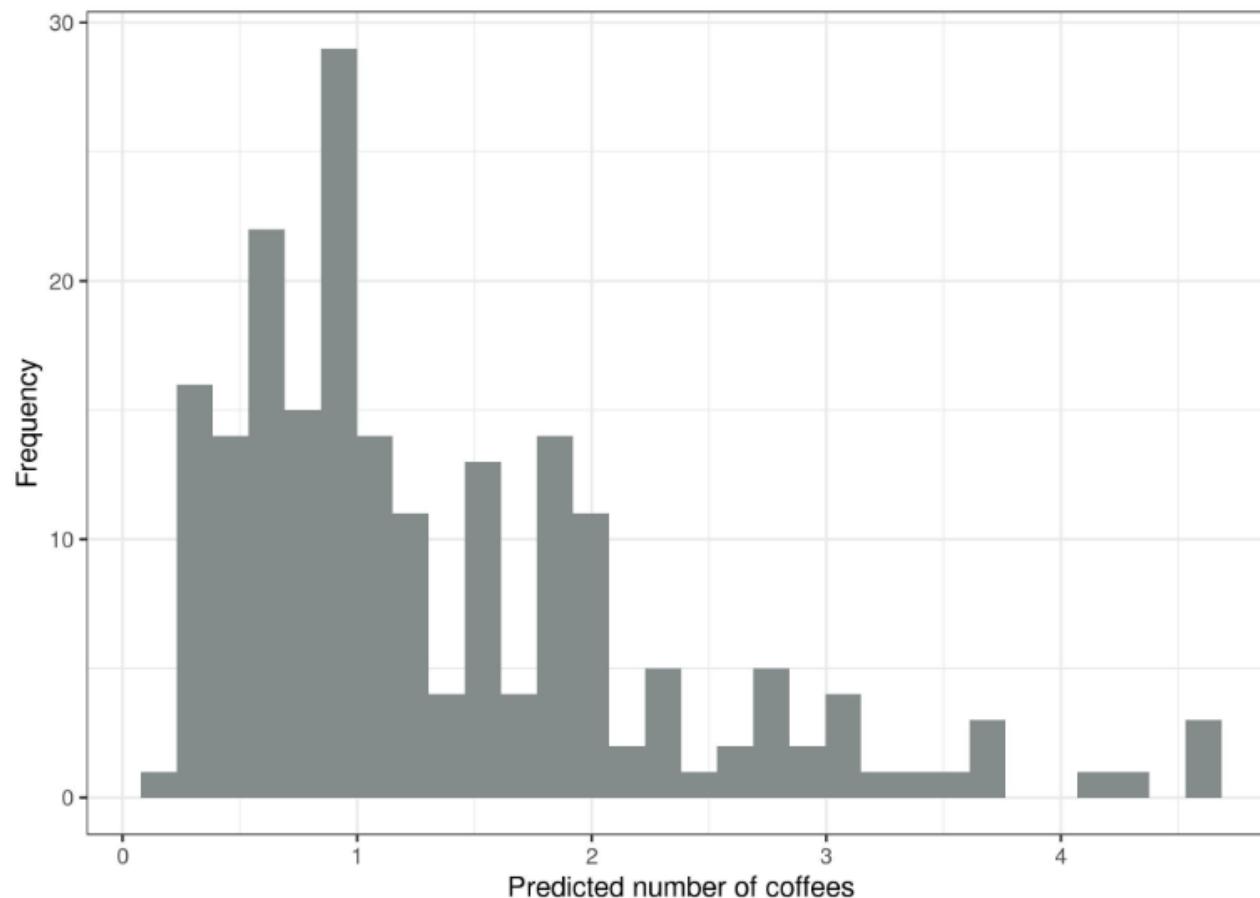
GLM

- ▶ Example: Number of coffees drank
 - ▶ Let's use OLS to predict the number of coffees drank
 - ▶ Problems:
 1. Negative predictions
 2. Invalidated coefficients and CIs of these (not shown here)



GLM

- ▶ Example: Number of coffees drank
 - ▶ Let's use Poisson Regression (GLM)



GLM

- ▶ Interpretation of GLM weights:
 - ▶ Example: Number of coffees drank

$$\ln(E(\text{coffee}|\text{str}, \text{slp}, \text{wrk})) = \beta_0 + \beta_{\text{str}}x_{\text{str}} + \beta_{\text{slp}}x_{\text{slp}} + \beta_{\text{wrk}}x_{\text{wrk}}$$

- ▶ Rewrite:

$$E(\text{coffee}|\text{str}, \text{slp}, \text{wrk}) = \exp(\beta_0 + \beta_{\text{str}}x_{\text{str}} + \beta_{\text{slp}}x_{\text{slp}} + \beta_{\text{wrk}}x_{\text{wrk}})$$

GLM

- ▶ Interpretation of GLM weights:

- ▶ Example: Number of coffees drank

$$E(\text{coffee}|\text{str}, \text{slp}, \text{wrk}) = \exp(\beta_0 + \beta_{\text{str}}x_{\text{str}} + \beta_{\text{slp}}x_{\text{slp}} + \beta_{\text{wrk}}x_{\text{wrk}})$$

- ▶ Results:

- ▶ Increasing the stress level by one point multiplies the expected number of coffees by the factor 1.12.
 - ▶ Increasing the sleep quality by one point multiplies the expected number of coffees by the factor 0.86.
 - ▶ The predicted number of coffees on a work day is on average 2.23 times the number of coffees on a day off.

	weight	exp(weight) [2.5%, 97.5%]
(Intercept)	-0.16	0.85 [0.54, 1.32]
stress	0.12	1.12 [1.07, 1.18]
sleep	-0.15	0.86 [0.82, 0.90]
workYES	0.80	2.23 [1.72, 2.93]

Interactions

- ▶ Linear regression model traditionally assumes no interactions between features
- ▶ Example: Number of rented bicycles
 - ▶ May be an interaction between temperature and whether it is a working day or not?
 - ▶ No need for new data, just do some feature engineering

Intercept	workY	temp	workY.temp
1	1	25	25
1	0	12	0
1	0	30	0
1	1	5	5

- ▶ The interaction of two categorical features works similarly.

Interactions

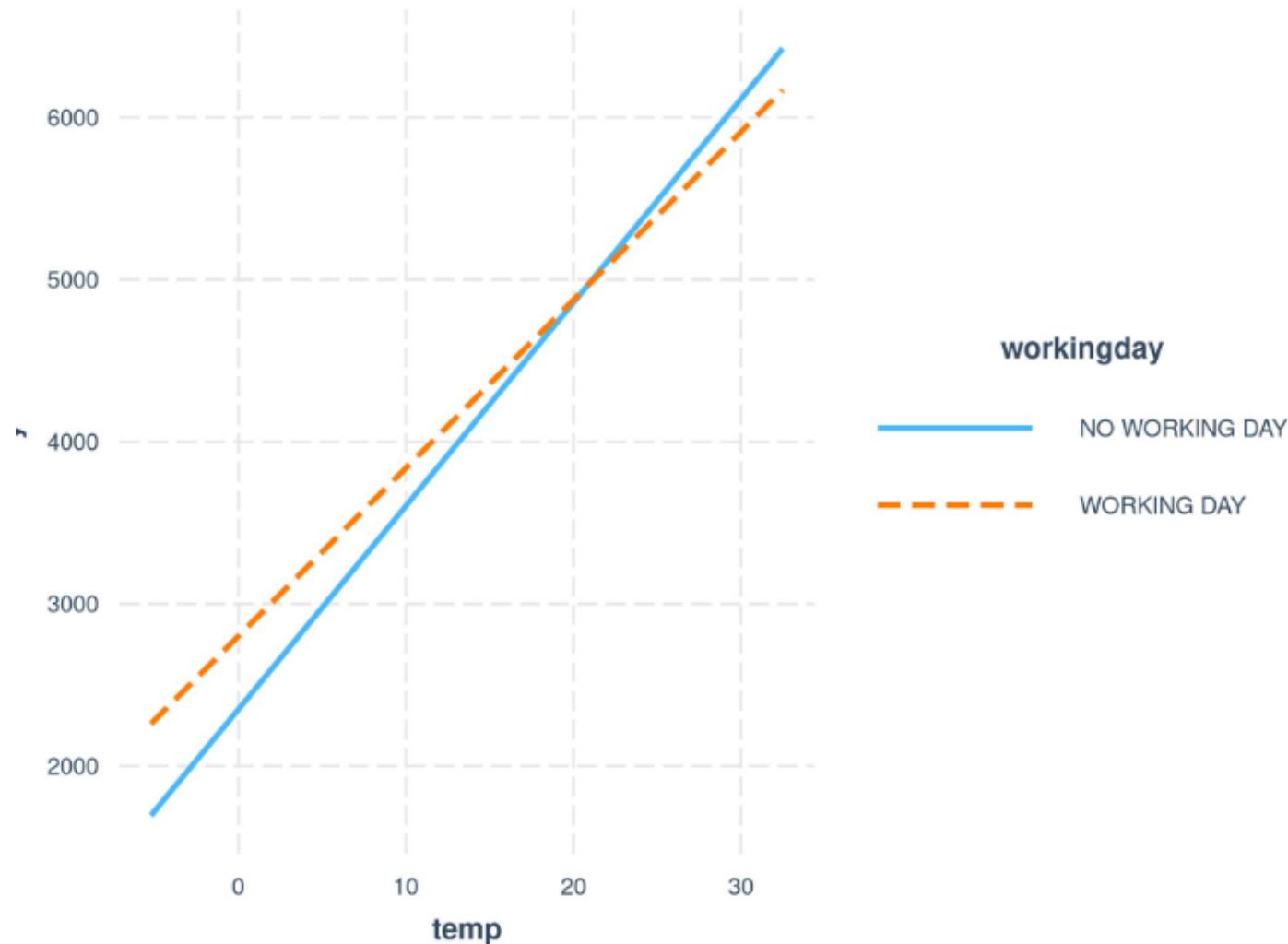
- ▶ Example: Number of bicycles rented
- ▶ Consider an interaction between the temperature and the working day

	Weight	Std. Error	2.5%	97.5%
(Intercept)	2185.8	250.2	1694.6	2677.1
seasonSPRING	893.8	121.8	654.7	1132.9
seasonSUMMER	137.1	161.0	-179.0	453.2
seasonFALL	426.5	110.3	209.9	643.2
holidayHOLIDAY	-674.4	202.5	-1071.9	-276.9
workingdayWORKING DAY	451.9	141.7	173.7	730.1
weathersitMISTY	-382.1	87.2	-553.3	-211.0
weathersitRAIN/...	-1898.2	222.7	-2335.4	-1461.0
temp	125.4	8.9	108.0	142.9
hum	-17.5	3.2	-23.7	-11.3
windspeed	-42.1	6.9	-55.5	-28.6
days_since_2011	4.9	0.2	4.6	5.3
workingdayWORKING DAY:temp	-21.8	8.1	-37.7	-5.9

Does the temperature have a negative effect given it is a working day?

Interactions

- ▶ Example: Number of bicycles rented

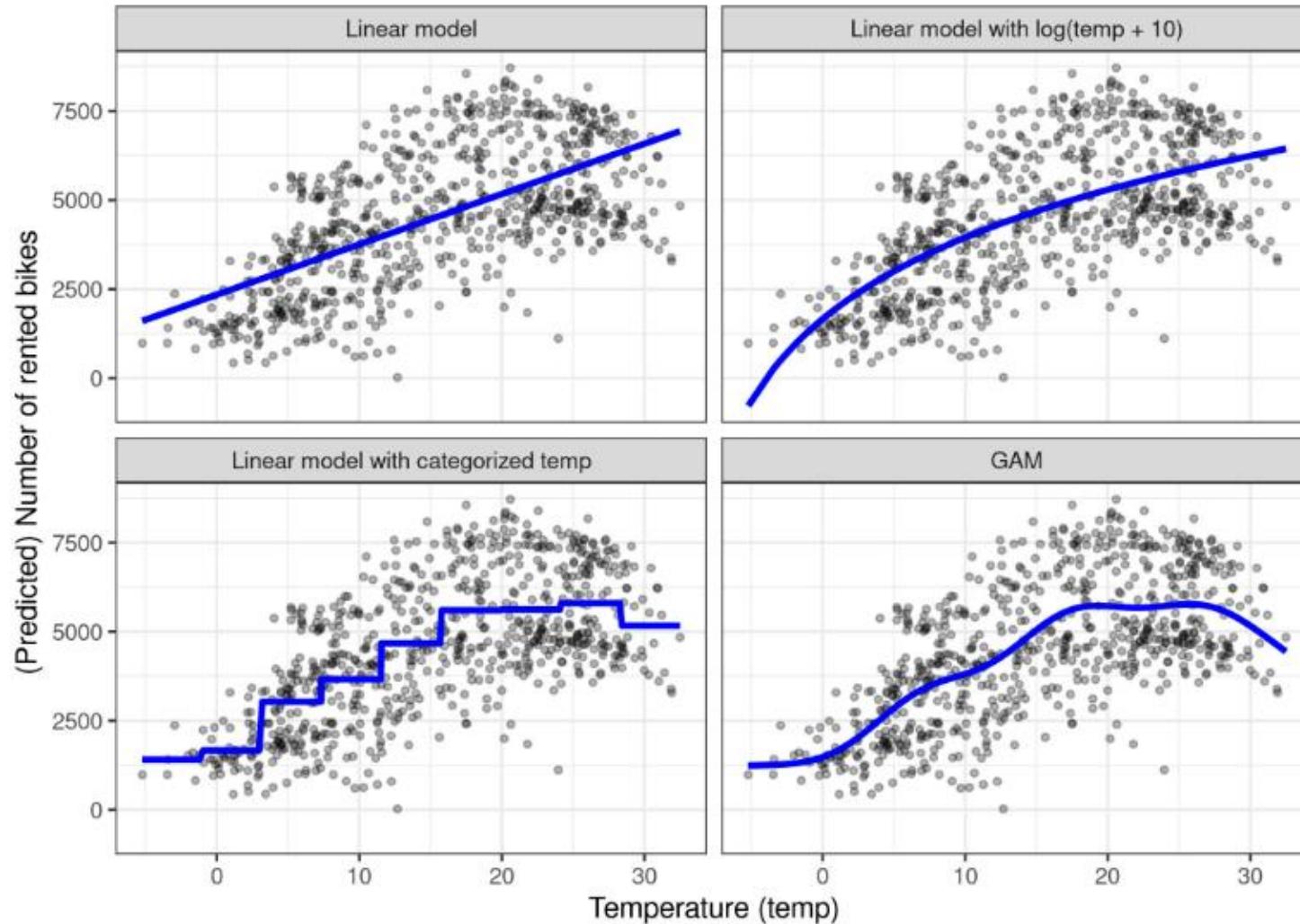


Nonlinear Effects: GAMs

- ▶ **The world is not linear.**
 - ▶ Increasing the temperature by one degree at 10° has the same effect on the number of rental bikes as increasing the temperature at 40° degrees?
 - ▶ The temperature feature has a linear, positive effect on the number of rental bikes, but at some point it flattens out and even has a negative effect at high temperatures.
- ▶ We can model nonlinear relationships via:
 1. Simple transformation of the feature (e.g. logarithm)
 2. Categorization of the feature
 3. Generalized Additive Models (GAMs)

GAM

▶ Example: Number of bicycles rented



GAM

- ▶ Linear Regression: Feature transformation
 - ▶ Logarithm is often used.
 - ▶ Every 10-fold temperature increase has the same linear effect on the number of bikes, so changing from 1 degree Celsius to 10 degrees Celsius has the same effect as changing from 0.1 to 1 (sounds wrong).
 - ▶ Other examples: square root, square, exponential
 - ▶ Trial or error or statistical packages (see BoxTidwell)
 - ▶ BEWARE: The interpretation of the feature changes according to the selected transformation.

GAM

- ▶ Linear Regression: Feature categorization:
 - ▶ Discretize the feature and turn it into a categorical feature.
 - ▶ Ex: Cut the temperature feature into 20 intervals with the levels [-10, -5), [-5, 0), ...
 - ▶ The linear model would estimate a step function because each level gets its own coefficient estimate.
 - ▶ BEWARE:
 - ▶ Need more data,
 - ▶ Possibility to overfit,
 - ▶ How to discretize the feature meaningfully (equidistant intervals or quantiles? how many intervals?).
 - ▶ Only use discretization if there is a very strong case for it (e.g., NA values for some instances)

GAM

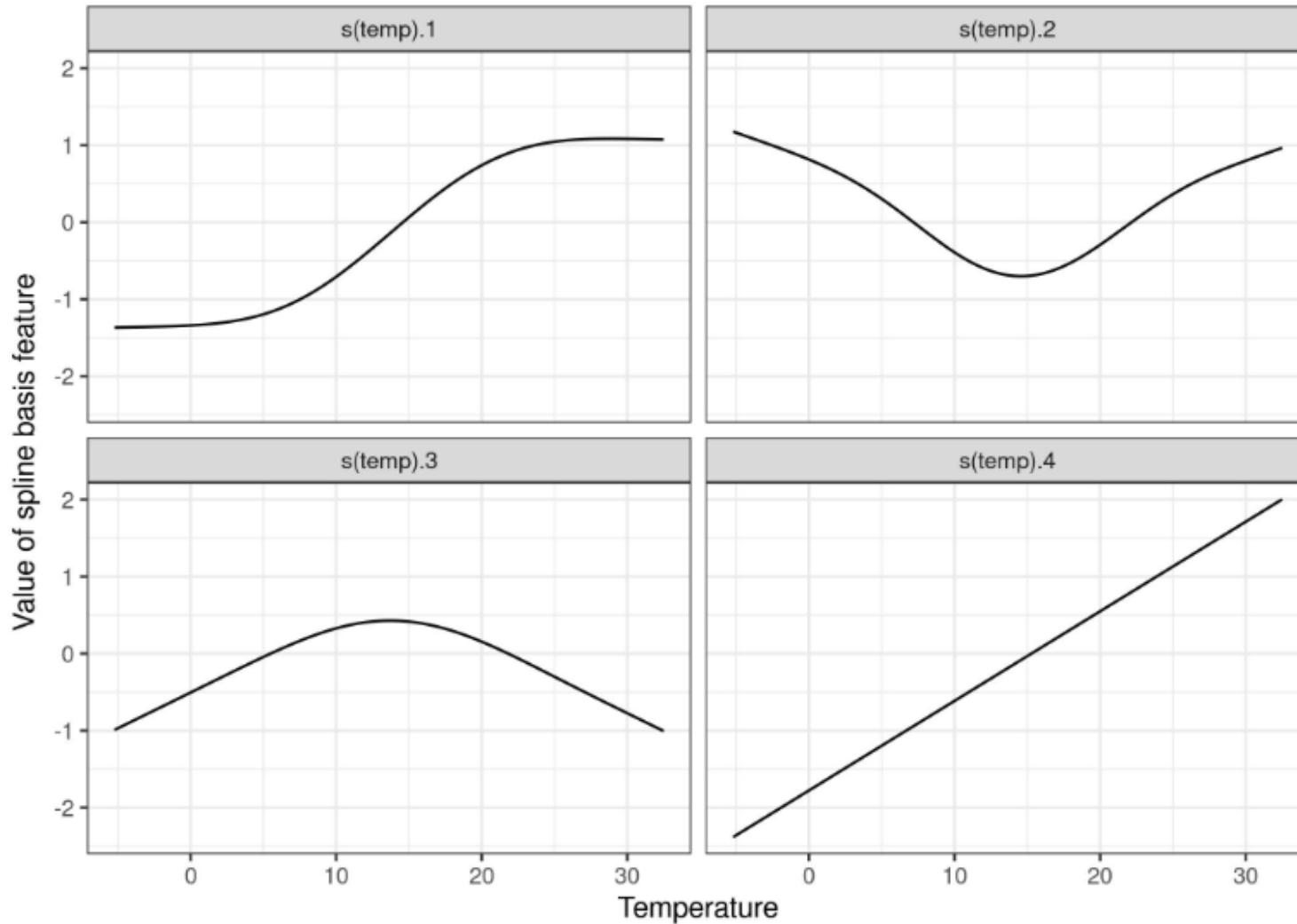
- ▶ GAMs assume that the outcome can be modeled by a sum of arbitrary functions of each feature.

$$g(E_Y(y|x)) = \beta_0 + f_1(x_1) + f_2(x_2) + \dots + f_p(x_p)$$

- ▶ Option to allow nonlinear relationships between some features and the output (which could still be linear for some features).
- ▶ How to learn these nonlinear functions $f_i()$?
 - ▶ Splines: Constructed from simpler basis functions
 - ▶ Could approximate more complex functions
 - ▶ Together with all linear effects, GAM also estimates weights for basis functions.
 - ▶ A penalty term for the weights to keep them close to zero to reduce overfitting.
 - ▶ A smoothness parameter to control the flexibility of the curve.

GAM

- ▶ Example: Number of bicycles rented



GAM

- ▶ Example: Number of bicycles rented
 - ▶ Values of spline basis functions for some of the instances in our dataset:

(Intercept)	s(temp).1	s(temp).2	s(temp).3	s(temp).4
1	-0.93	-0.14	0.21	-0.83
1	-0.83	-0.27	0.27	-0.72
1	-1.32	0.71	-0.39	-1.63
1	-1.32	0.70	-0.38	-1.61
1	-1.29	0.58	-0.26	-1.47
1	-1.32	0.68	-0.36	-1.59

GAM

- ▶ Example: Number of bicycles rented
 - ▶ GAM learns weights to each temperature spline basis feature

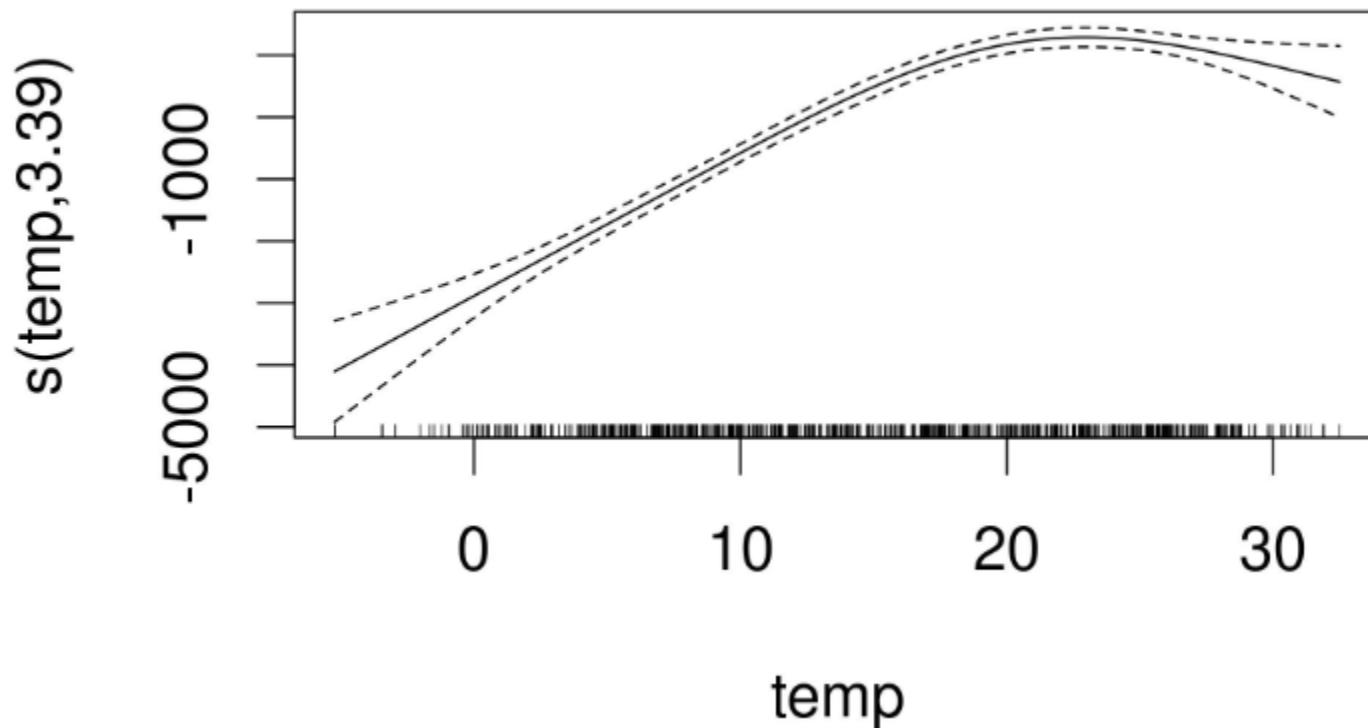
	weight
(Intercept)	4504.35
s(temp).1	989.34
s(temp).2	740.08
s(temp).3	2309.84
s(temp).4	558.27

GAM

- ▶ Example: Number of bicycles rented

- ▶ Resulting spline curve:

- ▶ At 0 degrees Celsius, the predicted number of bicycles is 3000 lower than the average prediction.



GLMs and GAMs

- ▶ Advantages:
 - ▶ Known and experienced within the community
 - ▶ Inference is a possibility: Confidence intervals for weights, significance tests, prediction intervals, etc.
 - ▶ The opacity of ML models arises due to:
 1. a lack of sparseness (many features are used,
 2. features are treated in a nonlinear fashion (more than one weight to describe the effect)
 3. modeling of interactions between the features.
 - ▶ GLM and GAM offer a good way to achieve a smooth transition to more flexible models, while preserving some of the interpretability.

GLMs and GAMs

- ▶ Disadvantages:
 - ▶ The sheer number of ways you can extend the simple linear model is overwhelming
 - ▶ Most modifications make the model less interpretable (unless the link function is identity)
 - ▶ The performance of tree-based ensembles like the random forest or gradient tree boosting may sometimes be better than the most sophisticated linear models.

GLMs and GAMs

- ▶ Further Extensions:
 - ▶ Problem: Data violates the assumption of being independent and identically distributed (e.g., repeated measurements on the same patient)
 - ▶ Solution: mixed models or generalized estimating equations
- ▶ Problem: Heteroscedastic errors
- ▶ Solution: Robust regression
- ▶ Problem: Outliers that strongly influence the model.
- ▶ Solution: Robust regression.
- ▶ Problem: Outcome to predict is a category (more than 2 categories)
- ▶ Solution: Multinomial regression.
- ▶ Problem: Want to predict ordered categories (e.g. ratings)
- ▶ Solution: Proportional odds model

GLMs and GAMs

- ▶ Further Extensions:
 - ▶ Problem: Outcome is a count (e.g.: number of children)
 - ▶ Solution: Poisson regression
- ▶ Problem: Same with above but the count value of 0 is very frequent
- ▶ Solution: Zero-inflated Poisson regression, hurdle model.
- ▶ Problem: Missing data.
- ▶ Solution: Multiple imputation
- ▶ Problem: Want to integrate prior knowledge into my models.
- ▶ Solution: Bayesian inference

ÖZYEĞİN ÜNİVERSİTESİ

DS 530

Fairness and Interpretability

ENİS KAYIŞ

Interpretable Models

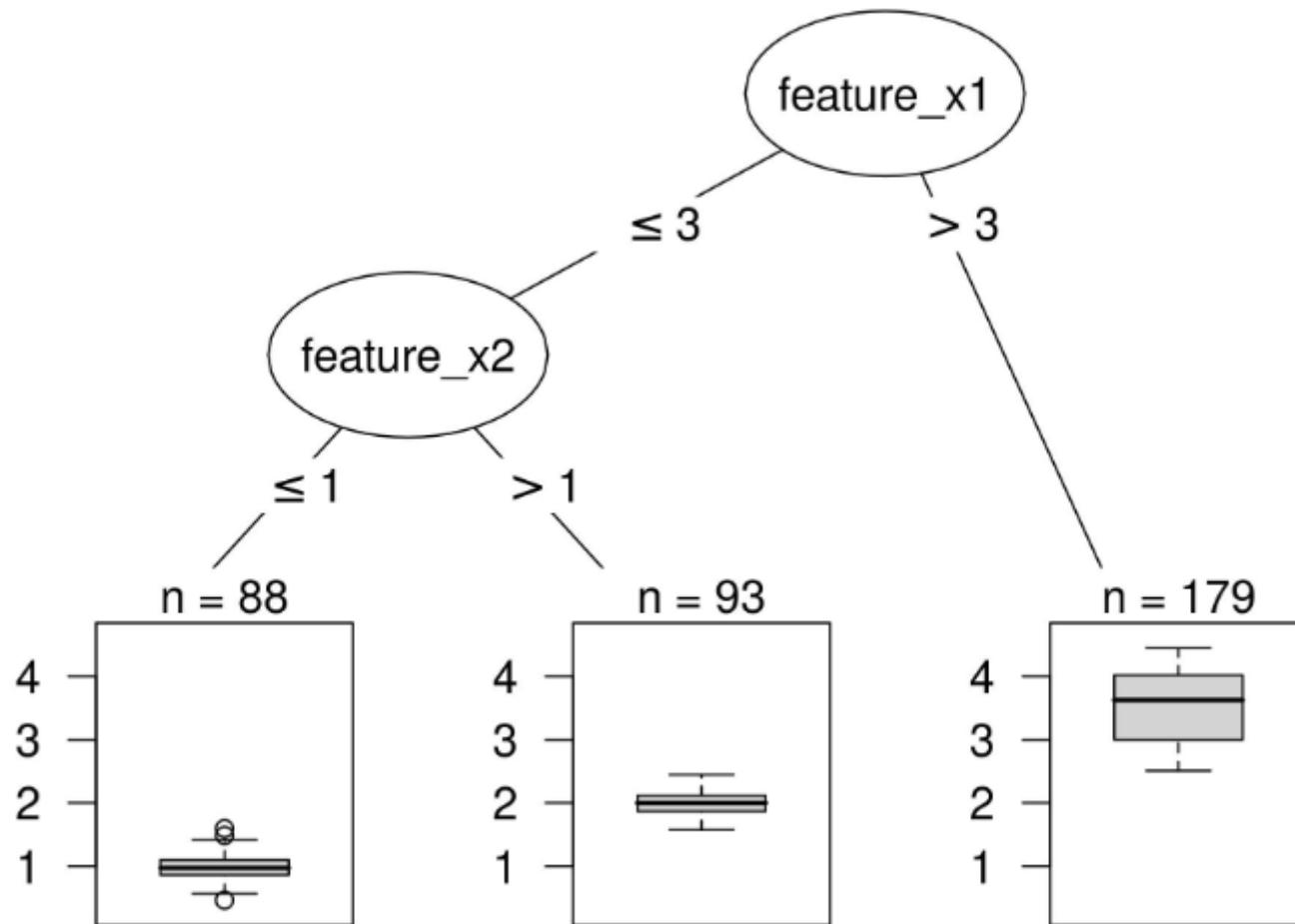
- ▶ Linear Regression
- ▶ Logistic Regression
- ▶ GLM, GAM and more
- ▶ Decision Tree
- ▶ Decision Rules
- ▶ RuleFit
- ▶ Others (Naïve Bayes, KNN)

Decision Tree

- ▶ Linear regression and logistic regression fail, where
 - ▶ the relationship between features and outcome is nonlinear
 - ▶ features interact with each other
- ▶ Basic idea:
 - ▶ Split the data multiple times using feature cutoff values
 - ▶ Subsets of the dataset, with each instance being in one subset
 - ▶ Prediction:
 - ▶ Regression Tree: The average outcome of the training data in this leaf
 - ▶ Classification Tree: The majority class of the training data in this leaf
- ▶ Many algorithms:
 - ▶ Myopic: **CART**, ID4, C4.5
 - ▶ Optimization Based: OCT, BinOCT, BendersOCT, RST
 - ▶ Changes: criteria for split, when to stop splitting, structure of the tree, etc.

Decision Tree

▶ Visual Example



Decision Tree

- ▶ Mathematically:

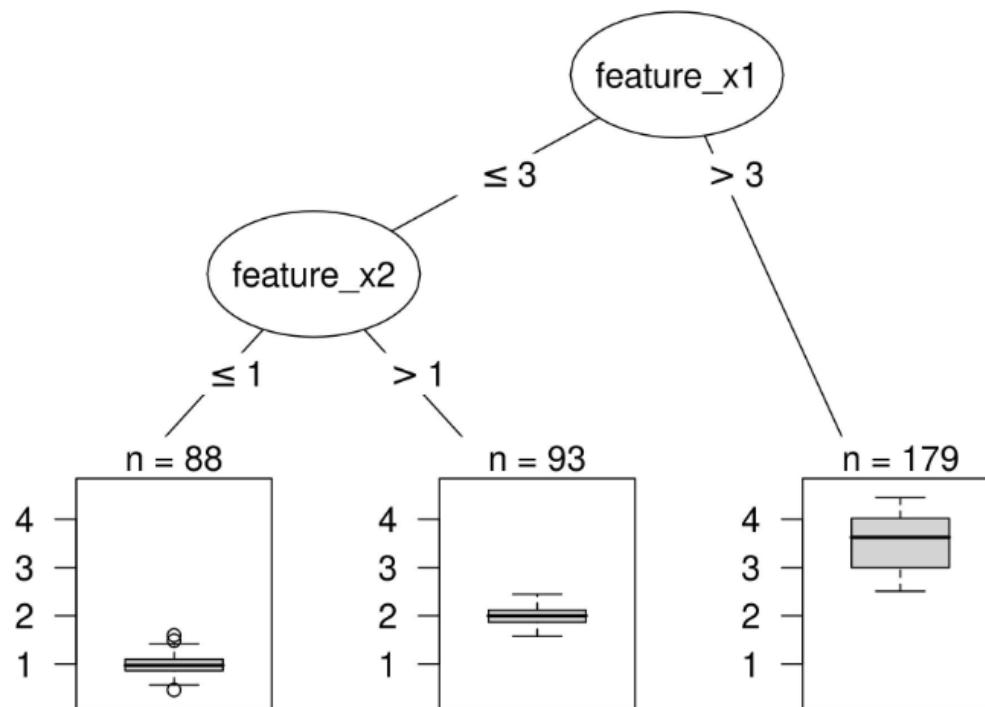
$$\hat{y} = \hat{f}(x) = \sum_{m=1}^M c_m I\{x \in R_m\}$$

- ▶ M leaf nodes (subsets R_m)
- ▶ c_m is the average outcome for all points in subset R_m
- ▶ Splitting criteria: Aim is to minimize variance of y in each leaf node
 - ▶ CART (sklearn) uses gini or entropy (classification) and MSE or MAE (regression)
 - ▶ Let p_{mk} be the proportion of class k observations in node m
 - ▶ Gini Impurity: $\sum_k p_{mk}(1 - p_{mk})$
 - ▶ Entropy: $-\sum_k p_{mk} \log_2(p_{mk})$

Decision Tree

▶ Interpretation:

- ▶ Simple, dictated by the splitting rules connected with “AND”
- ▶ If feature x_1 is less than or equal to 3 AND feature x_2 is more than 1, then the prediction is 2.



Decision Tree

- ▶ Feature Importance:
 - ▶ A measure of how much each feature has reduced the variance compared to the parent node.
- ▶ Tree Decomposition:
 - ▶ Individual predictions can be explained by decomposing the decision path into one component per feature

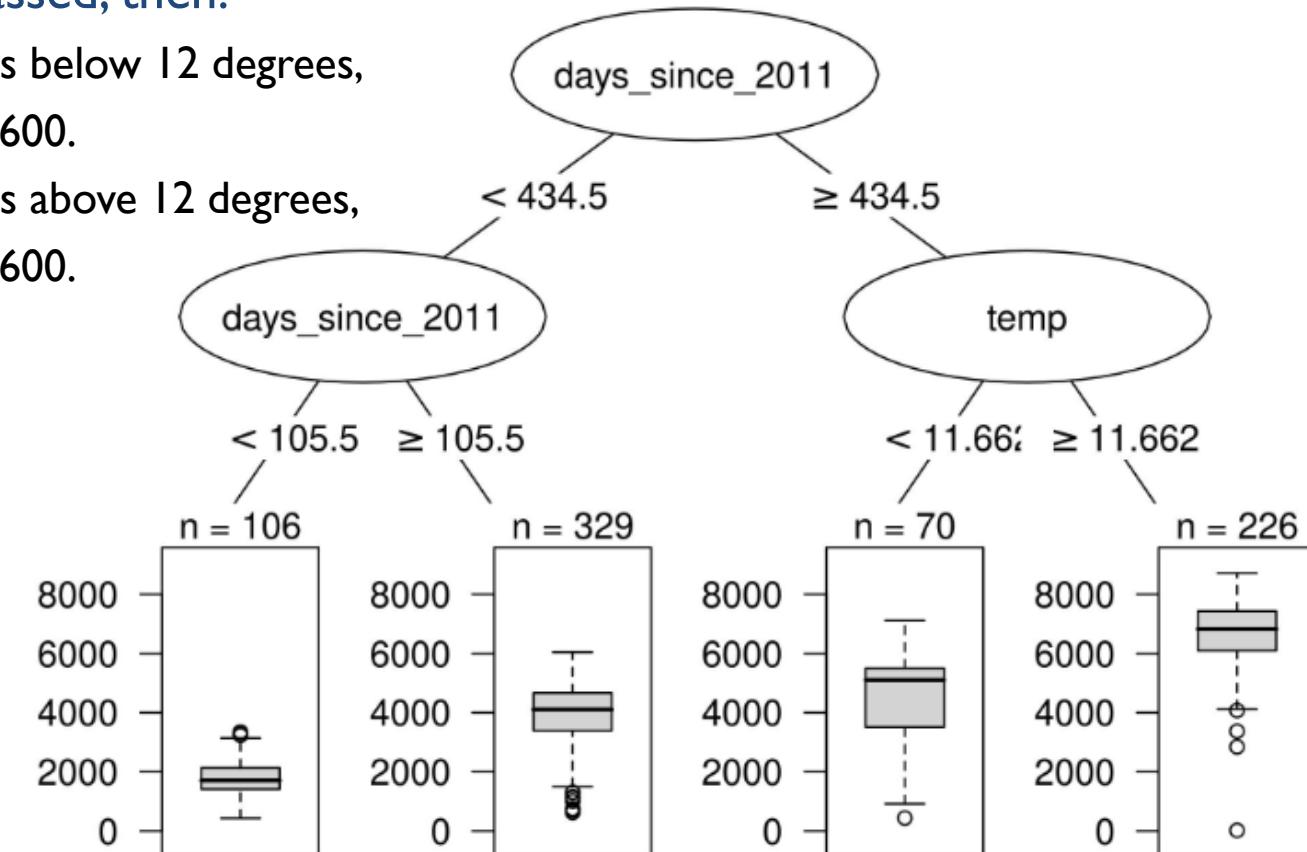
$$\hat{f}(x) = \bar{y} + \sum_{d=1}^D \text{split.contrib}(d, x) = \bar{y} + \sum_{j=1}^p \text{feat.contrib}(j, x)$$

- ▶ The mean of the target outcome + the sum of all contributions of the D splits that occur between the root node and the leaf node.
- ▶ A feature might be used for more than one split or not at all.
- ▶ Alternatively, add the contributions for each of the p features

Decision Tree

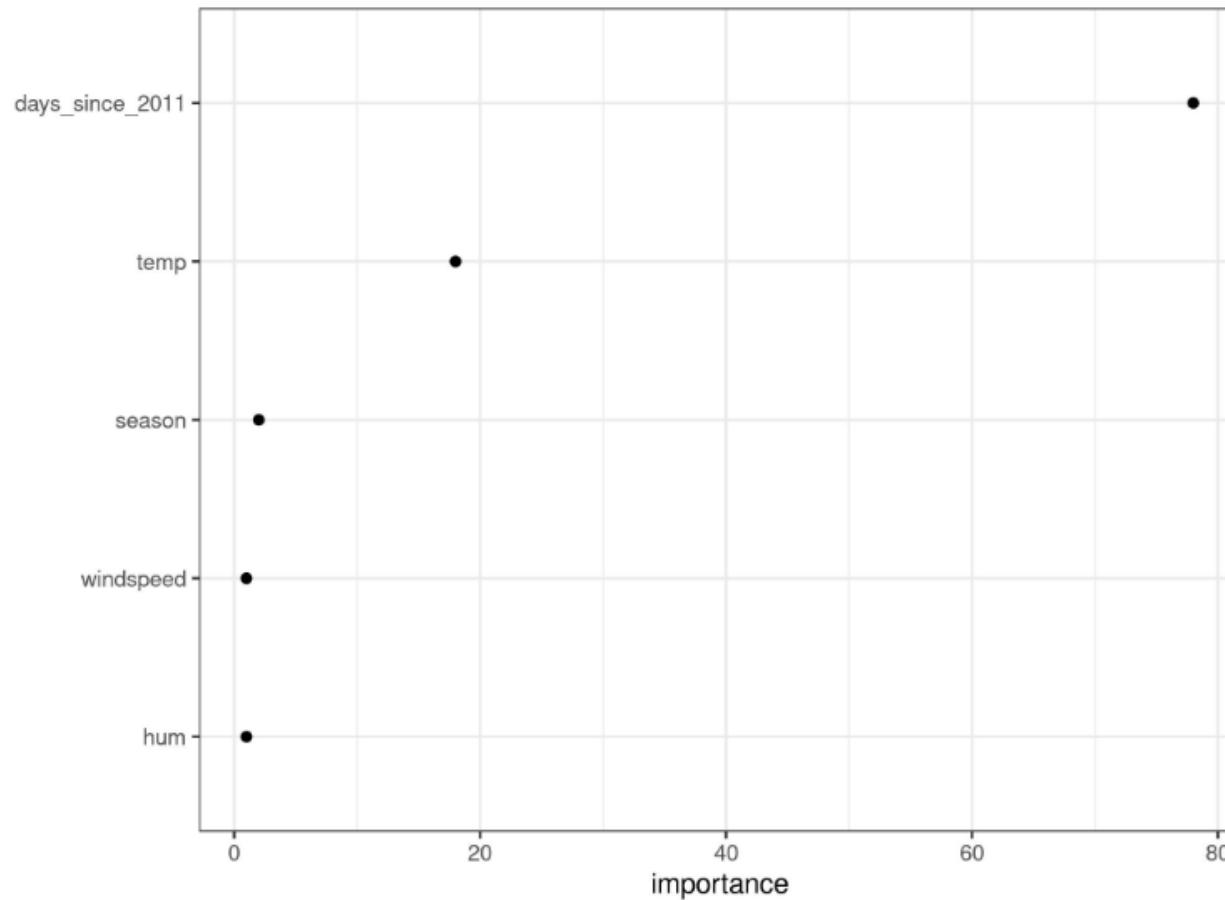
▶ Example: Bike Rentals Dataset

- ▶ If Days Passed ≤ 105 , the predicted number of bicycles is around 1800
- ▶ If $106 \leq \text{Days Passed} \leq 434$, it is around 3900.
- ▶ If $435 \leq \text{Days Passed}$, then:
 - ▶ If temperature is below 12 degrees, the prediction is 4600.
 - ▶ If temperature is above 12 degrees, the prediction is 6600.



Decision Tree

- ▶ Example: Bike Rentals Dataset
 - ▶ Feature Importance (note that this plot is created using a different tree than the one in the previous slide)

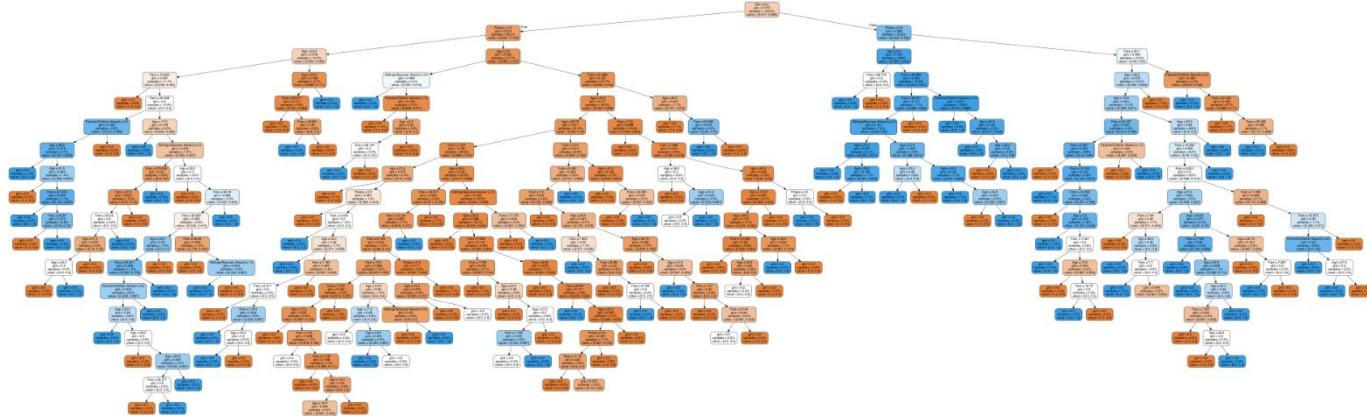


Decision Tree

- ▶ Advantages:
 - ▶ Can capture interactions between features
 - ▶ The interpretation is arguably pretty simple (distinct groups)
 - ▶ The tree structure also has a natural visualization
 - ▶ Trees create good explanations
 - ▶ Counterfactual: “If a feature had been greater/smaller than the split point, the prediction would have been y_1 instead of y_2 .”
 - ▶ Contrastive: Simply compare with the other leaf nodes of the tree
 - ▶ Selective: Better is the tree depth is small
 - ▶ Truthfulness: Depends on predictive performance
 - ▶ No need for transformation

Decision Tree

- ▶ Disadvantages:
 - ▶ Trees fail to deal with linear relationships.
 - ▶ Outcome has to be approximated by splits, creating a step function.
 - ▶ Lack of smoothness: Slight changes in the input feature can have a big impact on the predicted outcome
 - ▶ Instability: A few changes in the training dataset can create a completely different tree.
 - ▶ The number of terminal nodes increases quickly with depth.
 - ▶ Numer of leaf nodes for binary trees is 2^{depth}



Decision Rules

- ▶ Simple IF-THEN statement consisting of a condition and a prediction.
 - ▶ Ex: IF it rains today AND if it is April (condition), THEN it will rain tomorrow (prediction).
- ▶ A single decision rule or a combination of several rules can be used to make predictions.
- ▶ IF-THEN structure semantically resembles the way we think if:
 - ▶ the condition is built from intelligible features,
 - ▶ the length of the condition is short
 - ▶ there are not too many rules.
- ▶ Common in programming, but how do we learn them?

Decision Rules

- ▶ Ex: Predicting house value (low, medium, high)
- ▶ RULE: If a house is bigger than 100 square meters and has a garden, then its value is high.
 - ▶ size>100 is the first condition in the IF-part.
 - ▶ garden=1 is the second condition in the IF-part.
 - ▶ The two conditions are connected with an 'AND' to create a new condition. Both must be true for the rule to apply.
 - ▶ The predicted outcome (THEN-part) is value=high
- ▶ Not all rules are useful:
 - ▶ Support: The percentage of instances to which the condition of a rule applies
 - ▶ Ex: Rule (size=big AND location=good THEN value=high)
 - ▶ If 100 of 1000 houses are big and in a good location, then the support of the rule is 10%.
 - ▶ Accuracy: How accurate the rule is in predicting the correct class
 - ▶ Ex (ctd) Of the 100 houses with the above rule, 85 have value=high, 14 have value=medium and 1 has value=low, then the accuracy of the rule is 85%.
 - ▶ Usually there is a trade-off between accuracy and support:

Decision Rules

- ▶ Bad News: One rule does not suffice, generally. Need multiple (maybe 10 or 20) rules
- ▶ Problems:
 - I. Rules can overlap: Contradictory predictions
 - ▶ Combine rules:
 - Decision List (ordered)
 - Use the prediction of the first rule that applies
 - Decision Set (unordered)
 - Use the prediction of the majority (some might have higher voting power)
 - I. No rule applies: What is the prediction?
 - ▶ Use a default: The most frequent class of the data points which are not covered by other rules

Decision Rules

- ▶ How to learn decision rules?
- ▶ Many algorithms exist.
- ▶ **OneR:** Uses a single feature. Simple, interpretable, can be used as a benchmark.
- ▶ **Sequential covering:** Iteratively learns rules and removes the data points that are covered by the new rule.
- ▶ **Bayesian Rule Lists:** Combine pre-mined frequent patterns into a decision list using Bayesian statistics.

Decision Rules: OneR

- ▶ Selects the rule that carries the most information about the outcome
- ▶ BEWARE: Despite the name, it is actually one rule per unique feature value of the selected best feature. (More like OneFeatureRules)
- ▶ Algorithm:
 1. Discretize the continuous features by choosing appropriate intervals.
 2. For each feature:
 1. Create a cross table between the feature values and the (categorical) outcome.
 2. For each value of the feature, create a rule which predicts the most frequent class of the instances that have this particular feature value (can be read from the cross table).
 3. Calculate the total error of the rules for the feature.
 - ▶ Select the feature with the smallest total error.

Decision Rules: OneR

- ▶ What is the difference between decision trees?
- ▶ Ex: House Value Dataset

location	size	pets	value
good	small	yes	high
good	big	no	high
good	big	no	high
bad	medium	no	medium
good	medium	only cats	medium
good	small	only cats	medium
bad	medium	yes	medium
bad	small	yes	low
bad	medium	yes	low
bad	small	no	low

Decision Rules: OneR

- ▶ Ex: House Value Dataset:
 - ▶ Errors: location (4/10), size (3/10), pet (4/10)
 - ▶ Select size feature

	value=low	value=medium	value=high
size=big	0	0	2
size=medium	1	3	0
size=small	2	1	1

IF size=small THEN value=low
IF size=medium THEN value=medium
IF size=big THEN value=high

Decision Rules: OneR

- ▶ BEWARE:
 - ▶ OneR prefers features with many possible levels
 - ▶ Consider the extreme case: One instance per feature value
 - ▶ Thus, overfitting is easy. What to do?
 - ▶ Solution: Learn the rules on the training data and evaluate the total error on the validation set
 - ▶ Ties are resolved by taking the first feature!!!

Decision Rules: OneR

- ▶ Example: Cervical Cancer Dataset
 - ▶ All continuous input features were discretized into their 5 quantiles.
 - ▶ Age is chosen by OneR as the best predictive feature.
 - ▶ Any problems?

Age	prediction
(12.9,27.2]	Healthy
(27.2,41.4]	Healthy
(41.4,55.6]	Healthy
(55.6,69.8]	Healthy
(69.8,84.1]	Healthy

Decision Rules: OneR

- ▶ Imbalanced data: Very few cancer patients

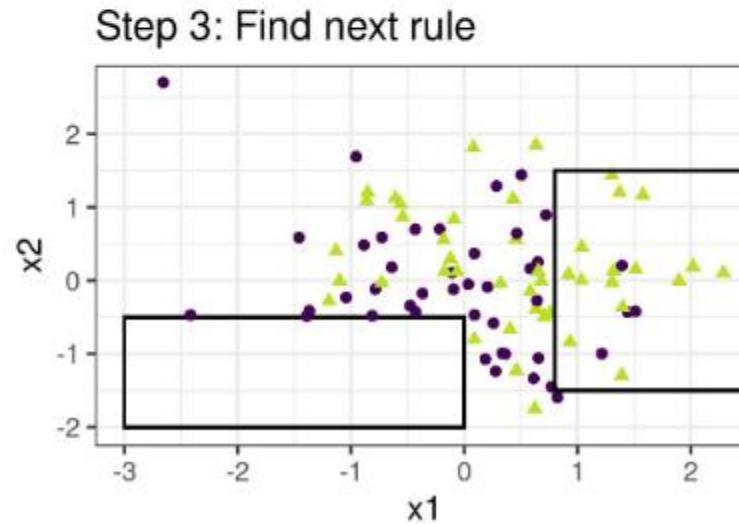
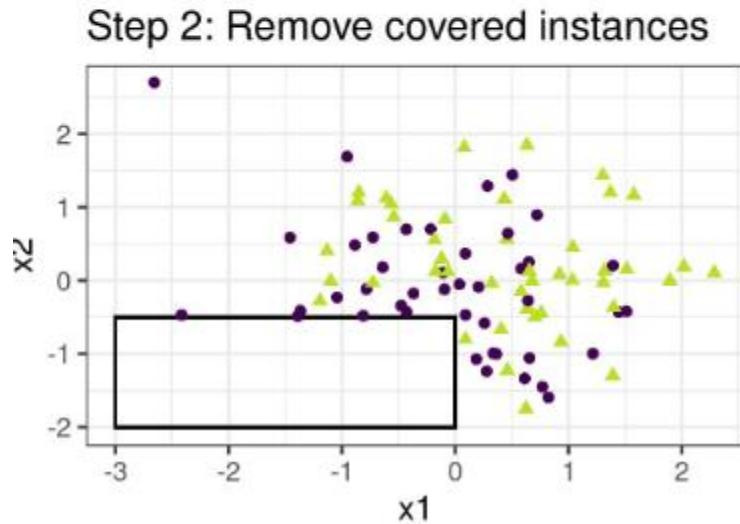
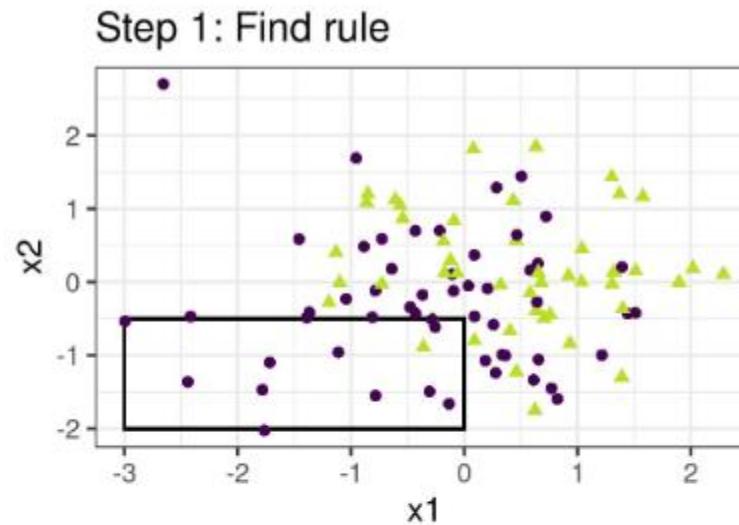
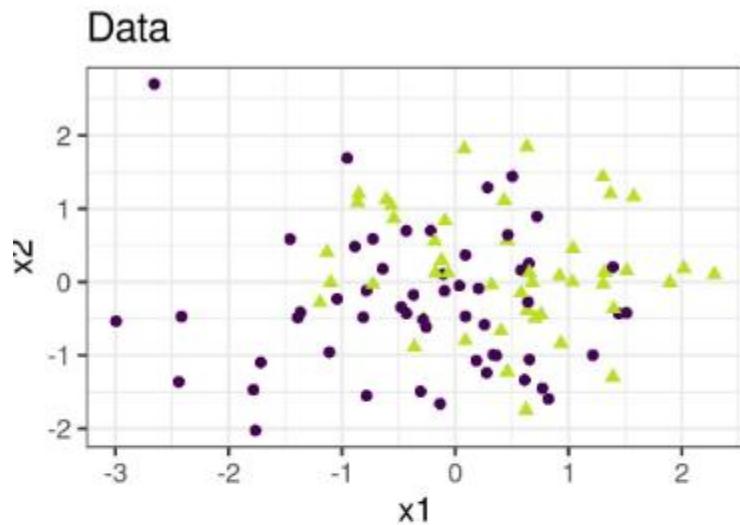
	# Cancer	# Healthy	P(Cancer)
Age=(12.9,27.2]	26	477	0.05
Age=(27.2,41.4]	25	290	0.08
Age=(41.4,55.6]	4	31	0.11
Age=(55.6,69.8]	0	1	0.00
Age=(69.8,84.1]	0	4	0.00

- ▶ Prediction for every feature-value is Healthy, thus the total error rate is the same for all features (55/858).
- ▶ The ties in the total error are, by default, resolved by using the first feature from the ones with the lowest error rates which happens to be the Age feature.
- ▶ OneR does not support regression tasks.
 - ▶ Turn a regression task into a classification task by cutting the continuous outcome into intervals

Decision Rules: Sequential Covering

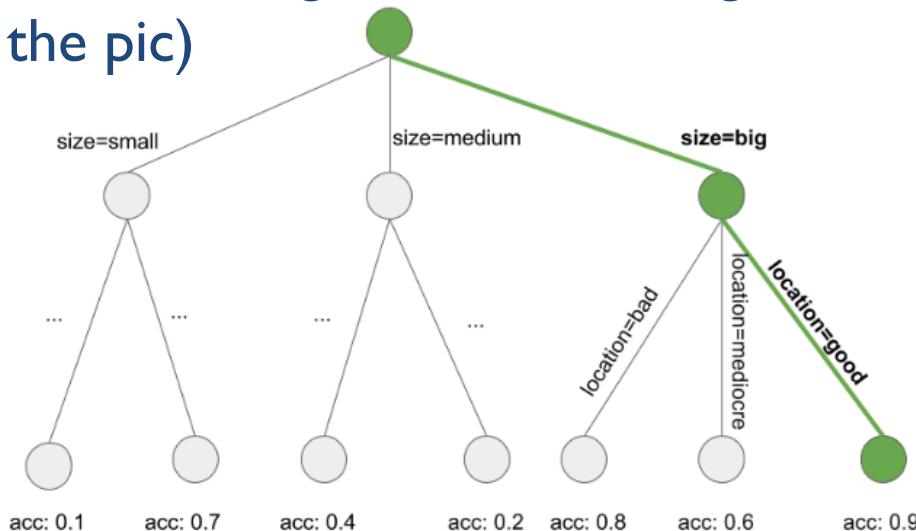
- ▶ Repeatedly learns a single rule to create a decision list that covers the entire dataset rule by rule.
- ▶ Many variants exist
- ▶ Idea: **Divide and conquer**
 1. Find a good rule that applies to some of the data points.
 2. Remove all data points which are covered by the rule (conditions), irrelevant of classification accuracy.
 3. Repeat the rule-learning and removal of covered points with the remaining points until no more points are left or another stop condition is met.

Decision Rules: Sequential Covering



Decision Rules: Sequential Covering

- ▶ How do we learn a single rule?
 - ▶ One Idea: learn a single rule from a decision tree
 1. Learn a decision tree (e.g., with CART)
 2. Start at the root node and recursively select the node with the lowest misclassification rate.
 3. The majority class of the terminal node is used as the rule prediction; the path leading to that node is used as the rule condition.
 - ▶ Ex: RULE: If location=good and size=big, then value=high (not shown in the pic)



Decision Rules: Sequential Covering

- ▶ Ex: Bike Rentals Dataset (notice that bike counts are turned into a categorical outcome using quartiles)

Decision List

rules

(temp >= 16) and (days_since_2011 <= 437) and (weathersit = GOOD) and (temp <= 24) and (days_since_2011 >= 131) => cnt=(4548,5956]

(temp <= 13) and (days_since_2011 <= 111) => cnt=[22,3152]

(temp <= 4) and (workingday = NO WORKING DAY) => cnt=[22,3152]

(season = WINTER) and (days_since_2011 <= 368) => cnt=[22,3152]

(hum >= 72) and (windspeed >= 16) and (days_since_2011 <= 381) and (temp <= 17) => cnt=[22,3152]

(temp <= 6) and (weathersit = MISTY) => cnt=[22,3152]

(hum >= 91) => cnt=[22,3152]

(mnth = NOV) and (days_since_2011 >= 327) => cnt=[22,3152]

(days_since_2011 >= 438) and (weathersit = GOOD) and (hum >= 51) => cnt=(5956,8714]

(days_since_2011 >= 441) and (hum <= 73) and (temp >= 15) => cnt=(5956,8714]

(days_since_2011 >= 441) and (windspeed <= 10) => cnt=(5956,8714]

(days_since_2011 >= 455) and (hum <= 40) => cnt=(5956,8714]

=> cnt=(3152,4548]

Decision Rules

- ▶ Advantages:
 - ▶ IF-THEN rules are easy to interpret (the most interpretable of the interpretable models, if the number of rules is small, the conditions of the rules are short, and the rules are ordered)
 - ▶ Decision rules can be **as expressive as decision trees, while being more compact.**
 - ▶ The **prediction with IF-THEN rules is fast**
 - ▶ Robust against monotonic transformations of the features and outliers
 - ▶ They **select only the relevant features** for the model.

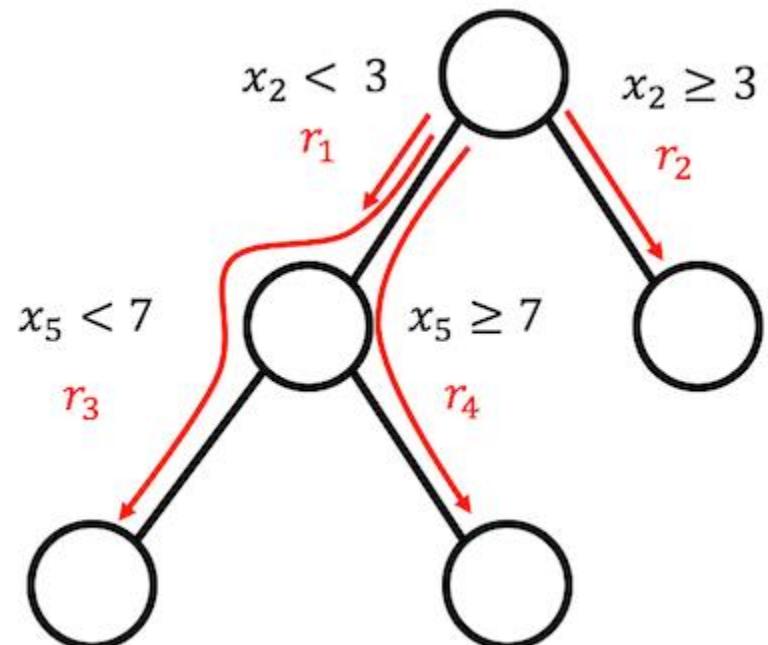
Decision Rules

- ▶ Disadvantages:
 - ▶ Focus on classification, almost completely neglects regression
 - ▶ Could turn any regression task into a classification problem, but could lose information
 - ▶ Often the features have to be categorical
 - ▶ Numeric features must be categorized
 - ▶ Decision rules are bad in describing linear relationships between features and output
 - ▶ Produce step-like prediction functions

RuleFit

- ▶ Learns sparse linear models that include automatically detected interaction effects (best of both worlds)
 - ▶ automatically generates them from decision trees
 - ▶ each rule is meaningful in predicting the outcome
 - ▶ How to select if many rules are available? (e.g., random forest)

- ▶ Ex:
 - ▶ r1: $x_2 < 3$
 - ▶ r2: $x_2 \geq 3$
 - ▶ r3: $x_2 < 3 \text{ & } x_5 < 7$
 - ▶ r4: $x_2 < 3 \text{ & } x_5 \geq 7$



RuleFit

► Ex: Bike Rentals

- ▶ Most important rule: “days_since_2011 > 111 & weathersit in (“GOOD”, “MISTY”)”
- ▶ Corresponding weight = 795: If the above rule applies, then the predicted number of bikes increases by 795, when all other feature values remain fixed.
- ▶ 278 such rules were created from the original 8 features!!!
- ▶ Thanks to Lasso, only 59 of the 278 have a nonzero weight.

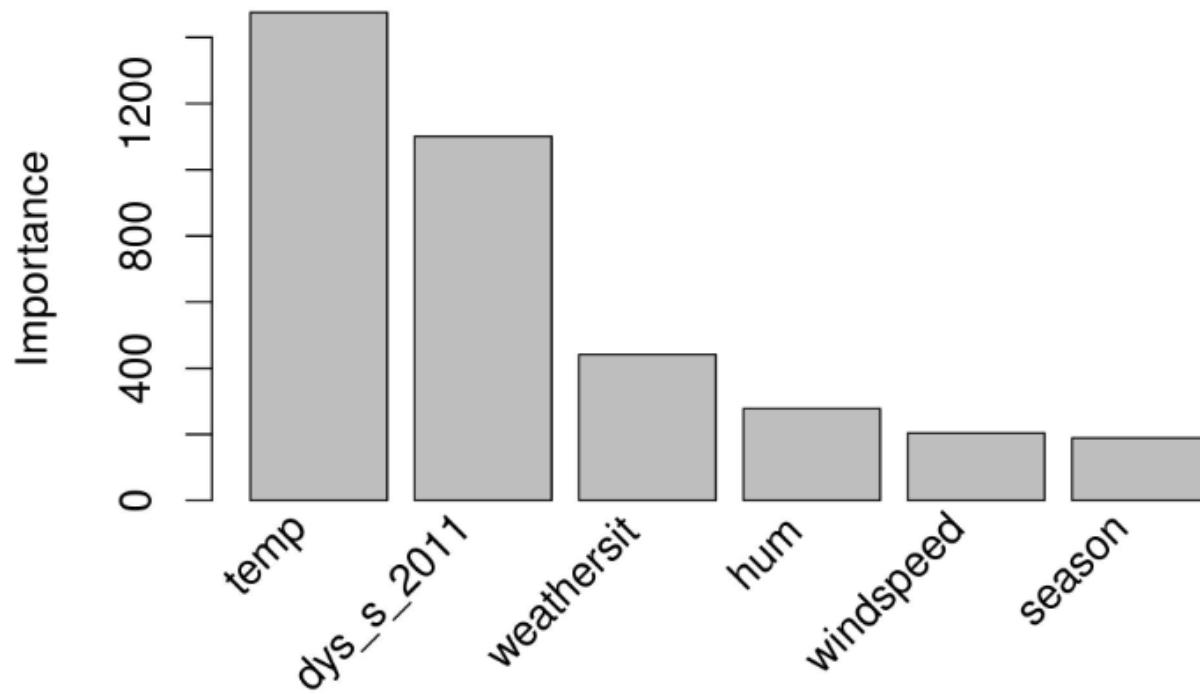
Description	Weight	Importance
days_since_2011 > 111 & weathersit in (“GOOD”, “MISTY”)	795	303
37.25 <= hum <= 90	-20	278
temp > 13 & days_since_2011 > 554	676	239
4 <= windspeed <= 24	-41	204
days_since_2011 > 428 & temp > 5	356	174

RuleFit

► Ex: Bike Rentals

- ▶ Temperature and time trend are the most important features
- ▶ Includes the importance of the raw term and all the decision rules with this feature

Variable importances



RuleFit

▶ Under the Hood:

▶ Step I: Rule generation

- ▶ Idea: Create a new set of binary features from your original features which can represent quite complex interactions of your original features.
- ▶ Creates “rules” from decision trees (simple or ensemble)
- ▶ Any path to a leaf node in a tree can be converted to a decision rule
- ▶ Try to generate a lot of diverse and meaningful rules
- ▶ Ensemble of trees, where each resulting tree is converted into multiple rules.

$$\hat{f}(x) = a_0 + \sum_{m=1}^M a_m \hat{f}_m(X)$$

- ▶ M is the number of trees
- ▶ $\hat{f}_m(x)$ is the prediction function of the m-th tree
- ▶ a_m is the weights of tree m

RuleFit

- ▶ Step I:
 - ▶ Each rule takes the form of:

$$r_m(x) = \prod_{j \in T_m} I(x_j \in s_{jm})$$

- ▶ T_m is the set of features in the m-th tree.
- ▶ Ex:

$$\begin{aligned} r_{17}(x) &= I(x_{\text{temp}} < 15) \cdot I(x_{\text{weather}} \in \{\text{good, cloudy}\}) \\ &\quad \cdot I(10 \leq x_{\text{windspeed}} < 20) \end{aligned}$$

- ▶ Even other rules from the same branch:

$$r_{18}(x) = I(x_{\text{temp}} < 15) \cdot I(x_{\text{weather}} \in \{\text{good, cloudy}\})$$

RuleFit

- ▶ Step 2:
 - ▶ MANY rules after step 1, now need to fit a model with reduced number of rules.
 - ▶ Every rule and every original feature becomes a feature in the linear model and gets a weight estimate.
 - ▶ The original raw features are added because trees fail at representing simple linear relationships between y and x .
 - ▶ First, winsorize original features (i.e., to remove possible outliers) and then normalize (equalize prior importance with the rules)

$$\hat{f}(x) = \hat{\beta}_0 + \sum_{k=1}^K \hat{\alpha}_k r_k(x) + \sum_{j=1}^p \hat{\beta}_j l_j(x_j)$$

- ▶ Lasso will choose the weights of alphas and betas.

RuleFit

▶ Advantages:

- ▶ Automatic addition of interactions (no need for a manual step)
- ▶ Can handle both classification and regression
- ▶ Good interpretability (assuming rules are short and there are not many of them)
 - ▶ Maybe work with small depth trees (~3)

▶ Disadvantages:

- ▶ Sometimes creates many rules with non-zero weights
- ▶ Interpretation with overlapping rules is not easy:
 - ▶ Ex: weight of “temp > 10” is 20 and weight of “temp > 15 & weather=‘GOOD’” is 25
 - ▶ Interpretation: Assuming **all other features remain fixed**, the predicted number of bikes increases by 25 when the weather is good and temperature above 15 degrees.
 - Problem: If the weather is good and the temperature is above 15 degrees, the temperature is automatically greater than 10.

Naive Bayes Classifier

- ▶ Uses the Bayes' theorem of conditional probabilities
- ▶ Naive Bayes is a conditional probability model and models the probability of a class C_k as:

$$P(C_k|x) = \frac{1}{Z} P(C_k) \prod_{i=1}^n P(x_i|C_k)$$

- ▶ Z is a scaling parameter such that probabilities sum to 1.
- ▶ The conditional probability of a class is the class probability times the probability of each feature given the class.
 - ▶ BEWARE: Very strong assumption on the conditional independence of the features!!!
- ▶ It can be interpreted on the modular level.
 - ▶ We know how much each feature contributes towards a certain class prediction.

K-Nearest Neighbors

- ▶ Uses the K nearest neighbors of a data point for prediction
 - ▶ Classification: assigns the most common class of the nearest neighbors
 - ▶ Regression: assign the average of the outcome of the neighbors
- ▶ Tricky: What is K? How to measure distance?
- ▶ Are predictions interpretable?
 - ▶ No parameters, thus no global model interpretability
 - ▶ Some local interpretability if one has few features and K is small.

ÖZYEĞİN ÜNİVERSİTESİ

DS 530

Fairness and Interpretability

ENİS KAYIŞ

Model Agnostic Methods

- ▶ Aim: To separate the explanations from the model
 - ▶ Flexibility: Works with any model
- ▶ Alternative: Use only interpretable models
 - ▶ Predictive performance may be lower
- ▶ Two groups:
 - ▶ Global methods: Describe how features affect the prediction on average
 - ▶ Local methods: Aim to explain individual predictions

Global Model-Agnostic Methods

- ▶ Describe the average behavior of a model
- ▶ Expressed as expected values based on the distribution of the data.
 - ▶ Ex: Partial dependence plot is the expected prediction when all other features are marginalized out.
- ▶ Useful when the modeler wants to understand the general mechanisms in the data or debug a model.
- ▶ Techniques:
 - ▶ Partial dependence plot: feature effect plot
 - ▶ Accumulated local effect plots: feature effect with dependent features
 - ▶ Feature interaction (H-statistic): strength of the joint effects of features
 - ▶ Functional decomposition: Decompose complex prediction function into smaller parts
 - ▶ Permutation feature importance: feature importance under permutation
 - ▶ Global surrogate models: Replaces the original model with a simpler model for interpretation

Partial Dependence Plot (PDP)

- ▶ Shows the marginal effect one or two features have on the predicted outcome of a model
 - ▶ Is the relationship linear, monotonic or more complex?
- ▶ Ex: Regression

$$\hat{f}_S(x_S) = E_{X_C} [\hat{f}(x_S, X_C)] = \int \hat{f}(x_S, X_C) d\mathbb{P}(X_C)$$

- ▶ x_S : Features for which PDP is plotted (usually 1 or 2, otherwise difficult to visualize)
 - ▶ x_C : Other features in the model (f)
- ▶ Works by marginalizing the model output over the distribution of the features in set C
- ▶ PDP is a function that depends only on features in S, interactions with other features included

Partial Dependence Plot (PDP)

- ▶ The partial function \hat{f}_S is estimated by calculating averages in the training data

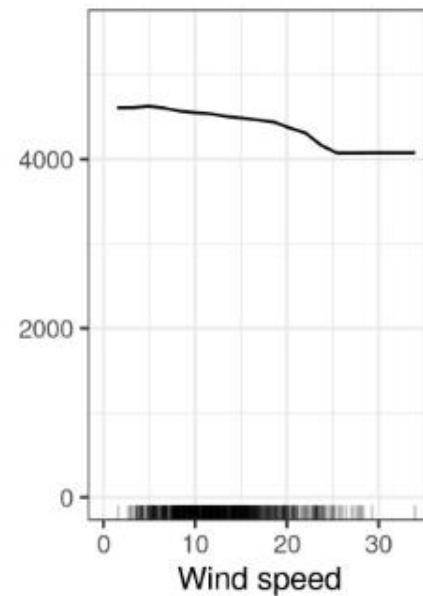
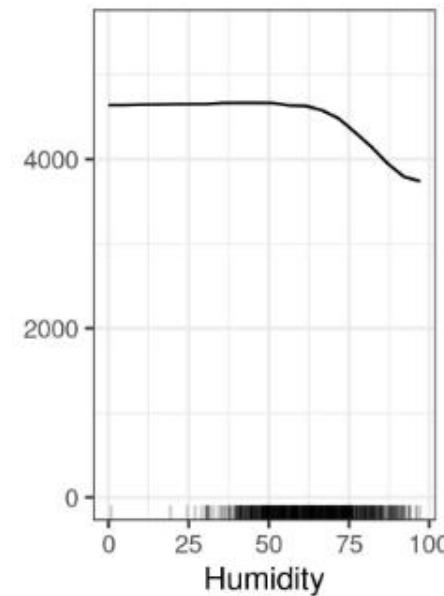
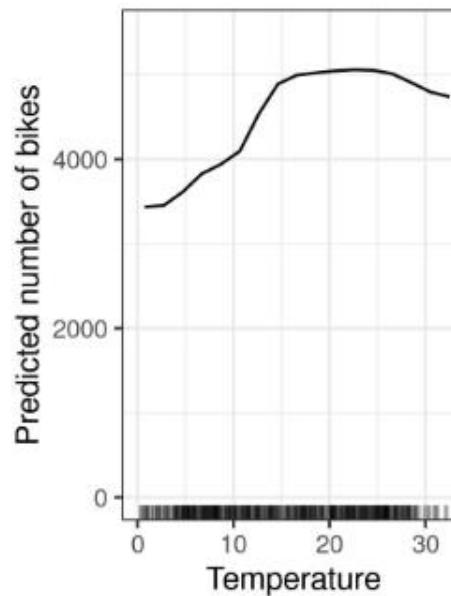
$$\hat{f}_S(x_S) = \frac{1}{n} \sum_{i=1}^n \hat{f}(x_S, x_C^{(i)})$$

- ▶ Assumption: The features in C are not correlated with the features in S.
- ▶ Classification: PDP displays the probability for a certain class given different values for feature(s) in S
- ▶ Categorical features:
 - ▶ For each category, force all datapoints to have the same category.
 - ▶ Ex (bike rentals): PDP for the season is a number for each season.

Partial Dependence Plot (PDP)

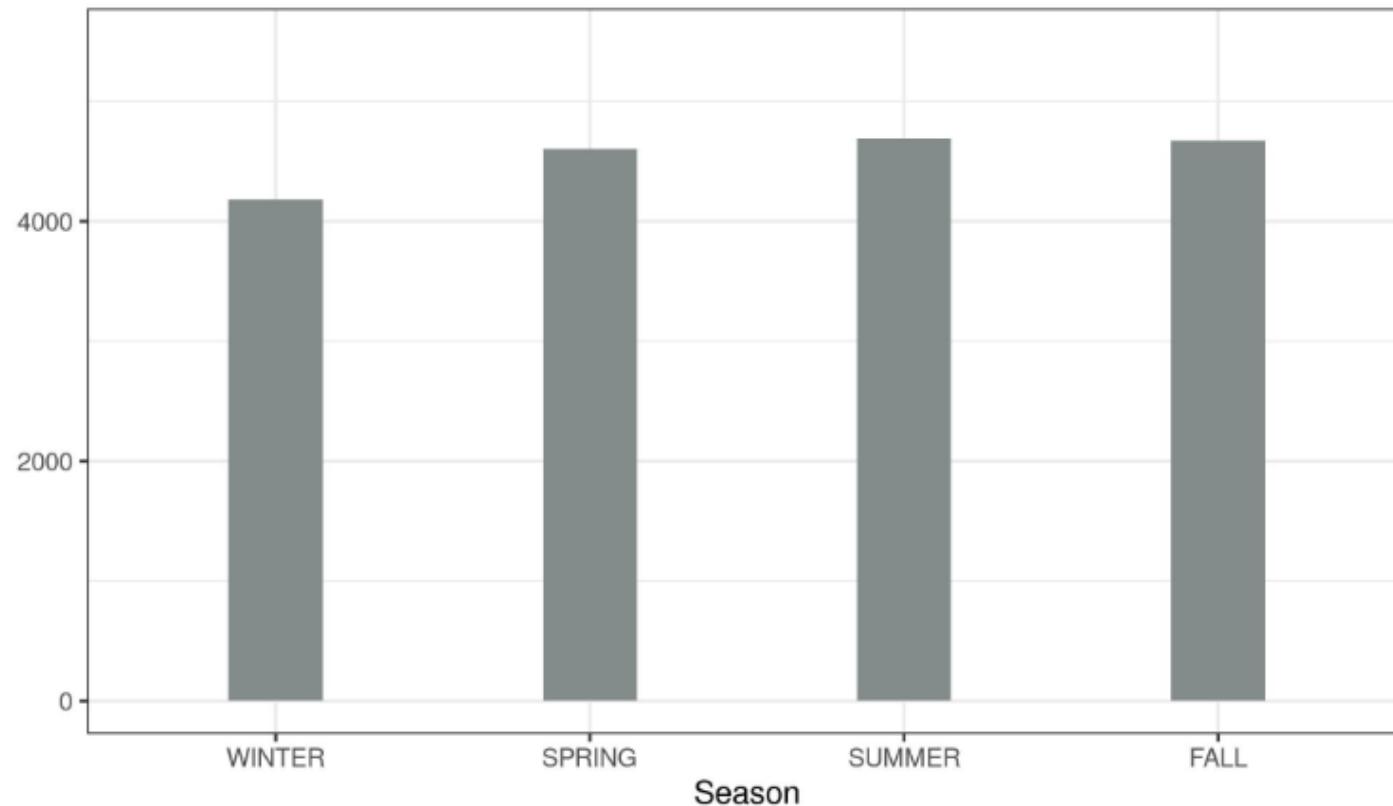
▶ Example: Bike Rentals Dataset

- ▶ Step 1: Fit a model and analyze PDPs
- ▶ In this example we use random forest as the learning model
- ▶ For warm but not too hot weather : On average, higher rentals
- ▶ Humidity exceeds 60%: On average less rentals
- ▶ More wind speed: the fewer people like to cycle
 - ▶ What happens when wind speed is between 25 to 35 km/h?



Partial Dependence Plot (PDP)

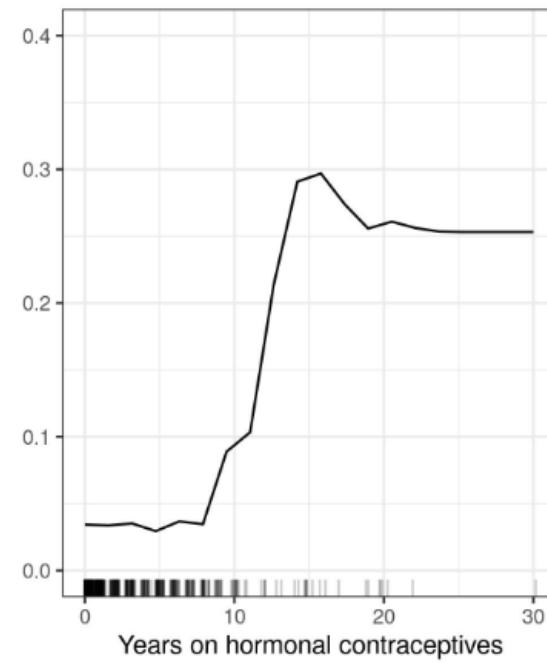
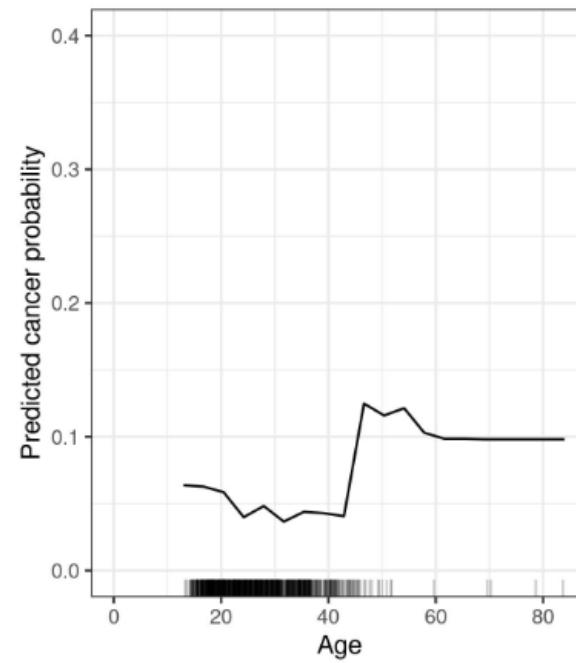
- ▶ Example: Bike Rentals Dataset
 - ▶ PDP for season
 - ▶ All seasons show similar effect on the model predictions, only for winter the model predicts fewer bicycle rentals. Interesting!



Partial Dependence Plot (PDP)

▶ Example: Cervical Cancer Dataset

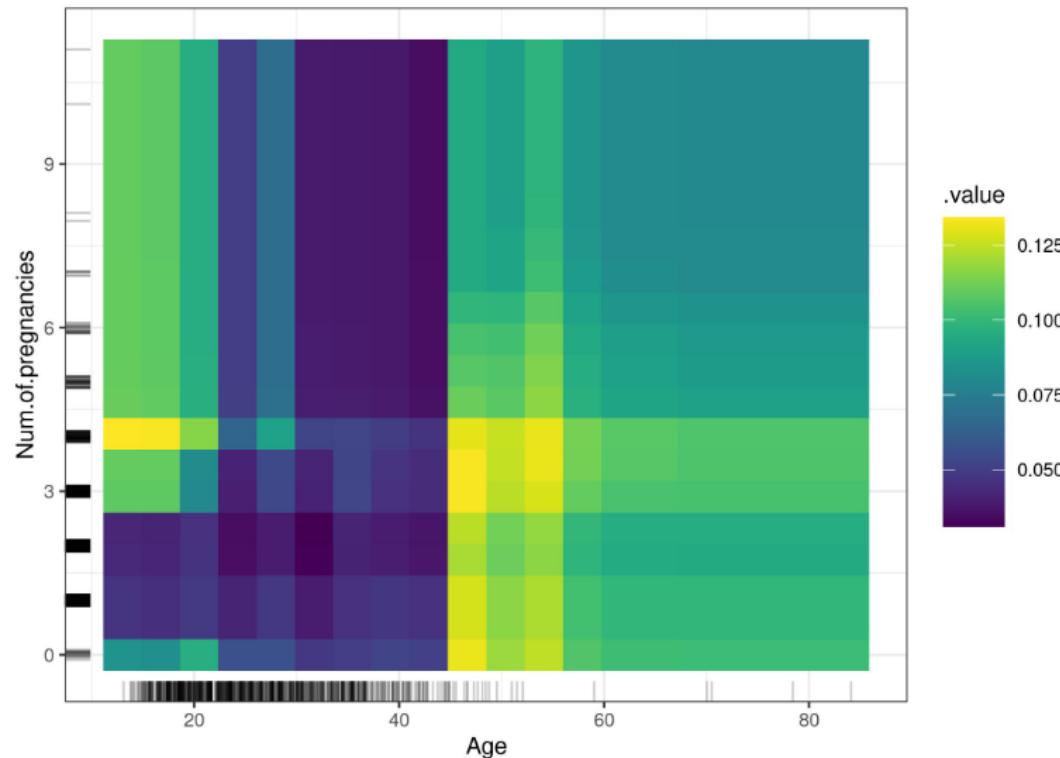
- ▶ PDP Plot using a trained RF model
- ▶ Age: the probability is low until 40 and increases after.
- ▶ Contraceptives: The more years, the higher the predicted cancer risk, especially after 10 years.
- ▶ Not many data points with large values (PD estimates are less reliable)



Partial Dependence Plot (PDP)

▶ Example: Cervical Cancer Dataset

- ▶ PDP Plot for pregnancy and age (using a trained RF model)
- ▶ Notice the increase in probability at 45.
- ▶ For ages < 25, women who had 1 or 2 pregnancies have a lower risk, compared with women who had 0 or more than 2 pregnancies.



Partial Dependence Plot (PDP)

- ▶ PDP-based Feature Importance
- ▶ Idea:
 - ▶ A flat PDP implies the feature is not important
 - ▶ The more the PDP varies, the more important the feature is.
 - ▶ For numerical features, it is the deviation of each unique feature value from the average curve.
- ▶ Beware:
 - ▶ Only captures the main effect, ignores interactions.
 - ▶ PDP could be flat as the feature affects the prediction mainly through interactions with other features

Partial Dependence Plot (PDP)

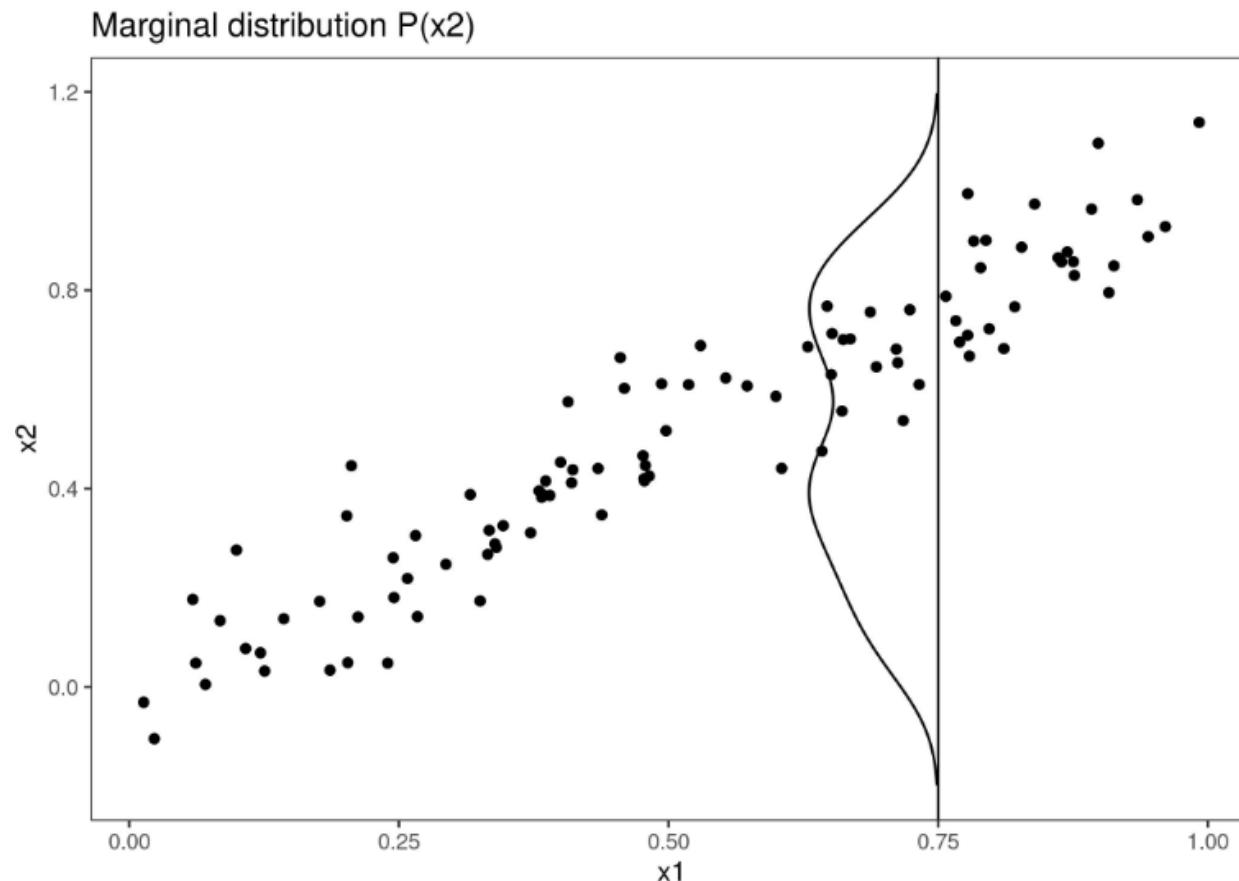
- ▶ Advantages:
 - ▶ Intuitive: Average prediction forcing all data points have that feature value
 - ▶ Clear Interpretation: Shows how the average prediction in your dataset changes when the j-th feature is changed
 - ▶ Easy to implement
 - ▶ Causal interpretation: We intervene on a feature and measure the changes in the predictions
- ▶ Disadvantages:
 - ▶ Realistic maximum number of features is two
 - ▶ Some PD plots do not show the feature distribution (what if no data points in some regions?)
 - ▶ Assumption of independence of features (see ALE plots)

Accumulated Local Effects (ALE)

- ▶ Describe how features influence the prediction **on average**
- ▶ Faster and unbiased alternative to PDPs
 - ▶ With correlated features, PDPs cannot be trusted.
 - ▶ Great bias in the estimated feature effect
 - ▶ Example:
 - ▶ To calculate PDP for, say 30 m², size of a house, PDP replace the living area for all instances by 30 m²
 - ▶ Even for houses with 10 rooms!!!
 - ▶ The partial dependence plot includes these unrealistic houses in the feature effect estimation and pretends that everything is fine.

ALE Plot

Example for why PDP is not realistic with strongly correlated features x_1 and x_2 .

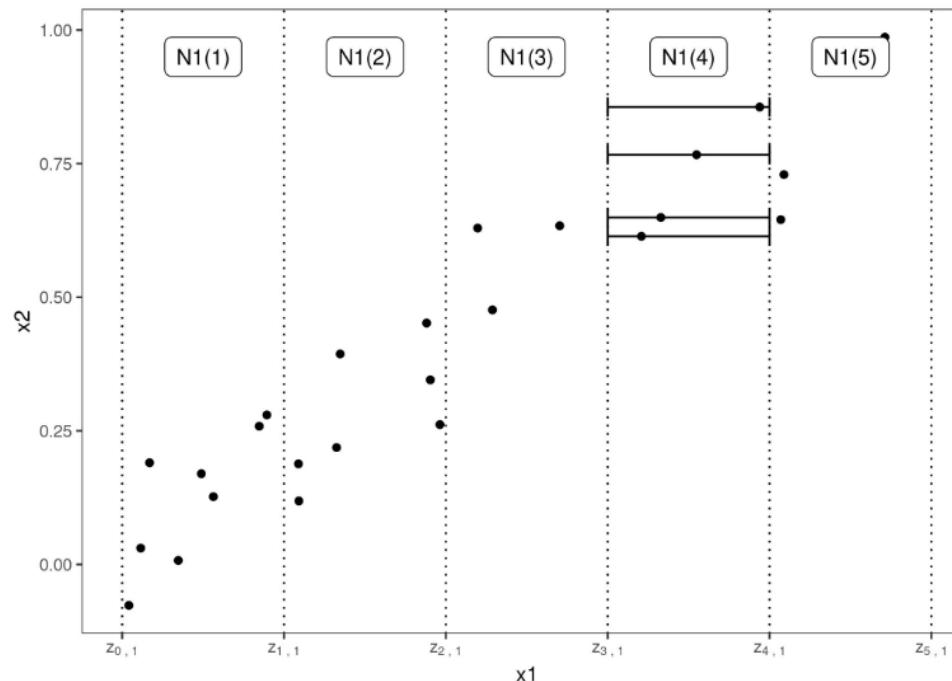


ALE Plot

- ▶ ALE plots solve correlated features problem by calculating differences in predictions instead of averages.
- ▶ For the effect of living area at 30 m², ALE uses all houses with **about** 30 m², gets the model predictions pretending these houses were 31 m² minus the prediction pretending they were 29 m².
- ▶ This gives us the pure effect of the living area and is not mixing the effect with the effects of correlated features.

ALE Plot

- ▶ Calculation of ALE for feature x_1 , which is correlated with x_2 .
 - ▶ Step 1: Divide the feature into intervals (use quantiles)
 - ▶ Step 2: For the data instances in an interval, calculate the difference in the prediction when we replace the feature with the upper and lower limit of the interval.
 - ▶ Step 3: Accumulate and center the differences



ALE Plot

▶ Estimation:

- ▶ Divide the feature into many intervals and compute the differences in the predictions
- ▶ Uncentered Effect:

$$\hat{f}_{j,ALE}(x) = \sum_{k=1}^{k_j(x)} \frac{1}{n_j(k)} \sum_{i: x_j^{(i)} \in N_j(k)} \left[\hat{f}(z_{k,j}, x_{-j}^{(i)}) - \hat{f}(z_{k-1,j}, x_{-j}^{(i)}) \right]$$

Accumulated
Local
Effect

- ▶ Then, we center this value so that the mean effect is zero:

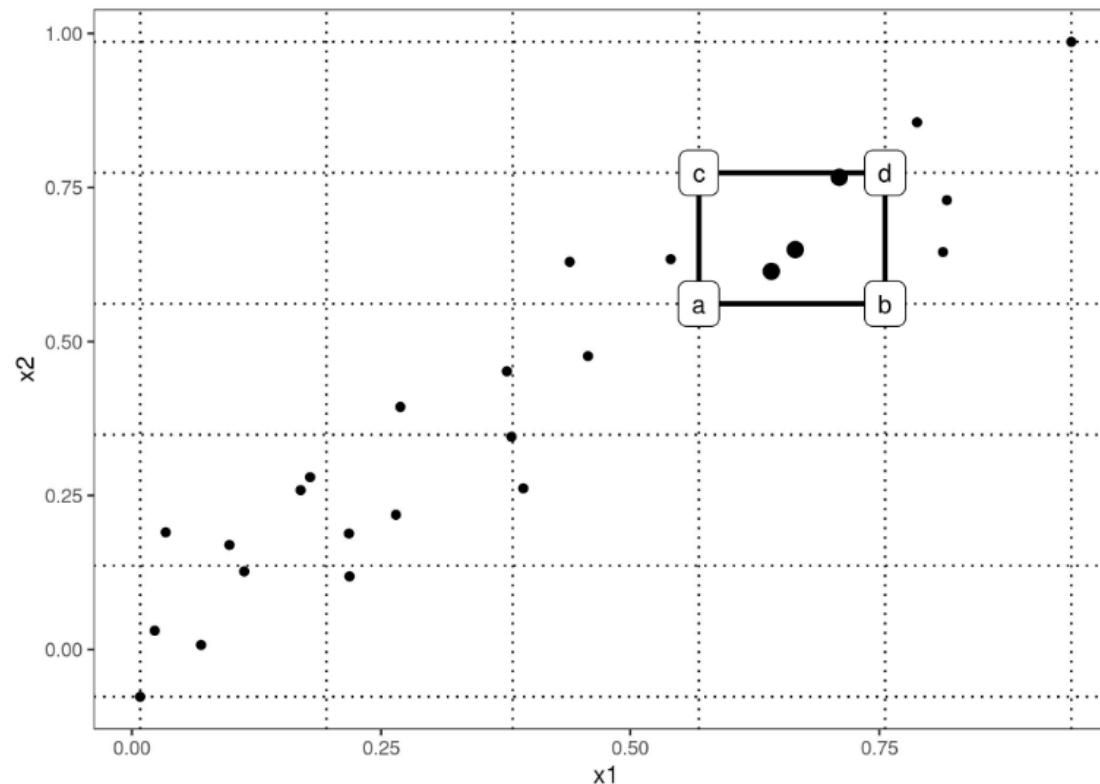
$$\hat{f}_{j,ALE}(x) = \hat{f}_{j,ALE}(x) - \frac{1}{n} \sum_{i=1}^n \hat{f}_{j,ALE}(x_j^{(i)})$$

ALE Plot

- ▶ Interpretation:
 - ▶ The main effect of the feature at a certain value compared to the average prediction of the data
 - ▶ Ex: ALE estimate of -2 at $x_j = 3$ means:
 - ▶ If the j-th feature has value 3, then the prediction is lower by 2 compared to the average prediction

ALE Plot

- ▶ Interaction of two features
 - ▶ Only shows the additional interaction effect of the two features
 - ▶ (1) replace values of x_1 and x_2 with the values from the cell corners, then (2) calculate 2nd-order difference is $(d - c) - (b - a)$, then (3) the mean 2nd-order difference in each cell is accumulated over the grid and centered.

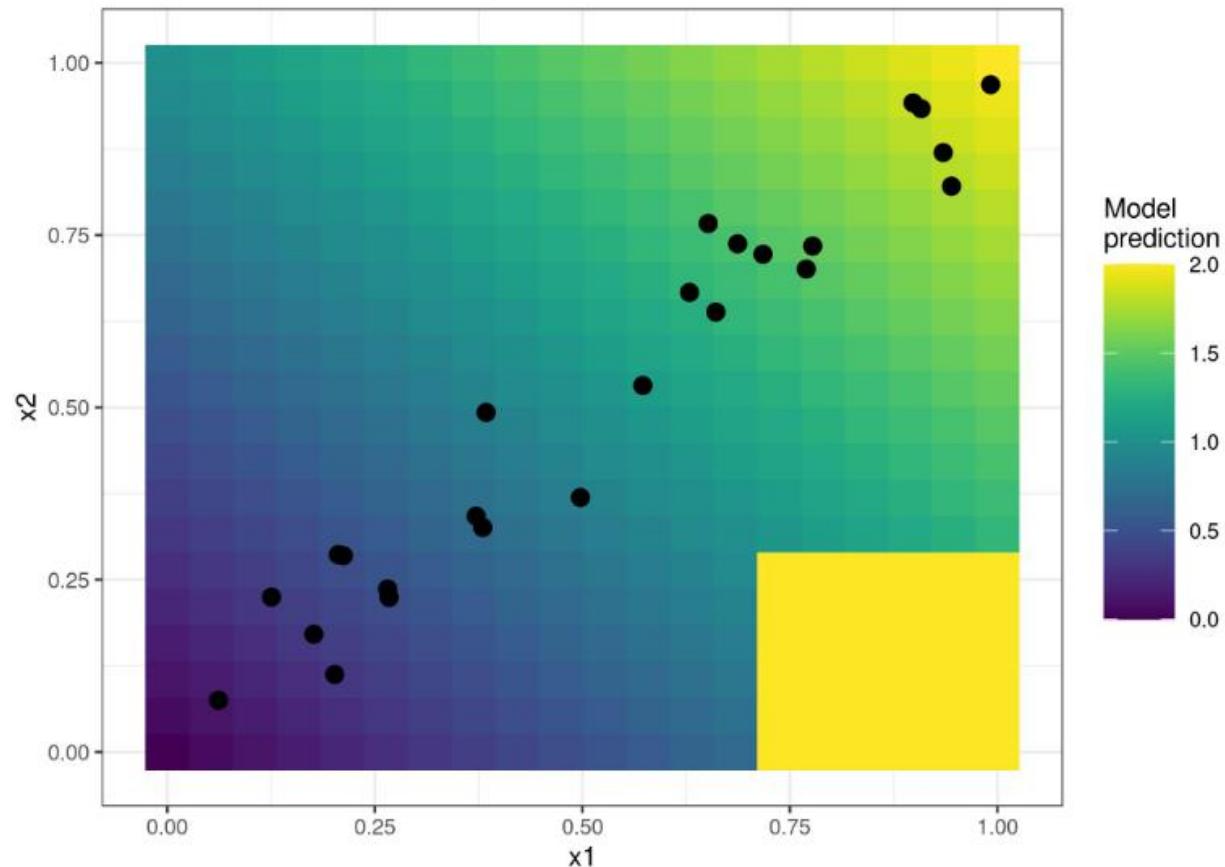


ALE Plot

- ▶ Interpretation:
 - ▶ The second-order effect is the additional interaction effect of the features **after we have accounted for the main effects.**
 - ▶ ALE plots and PD plots differ:
 - ▶ PDPs always show the total effect
 - ▶ ALE plots show the first- or second-order effect

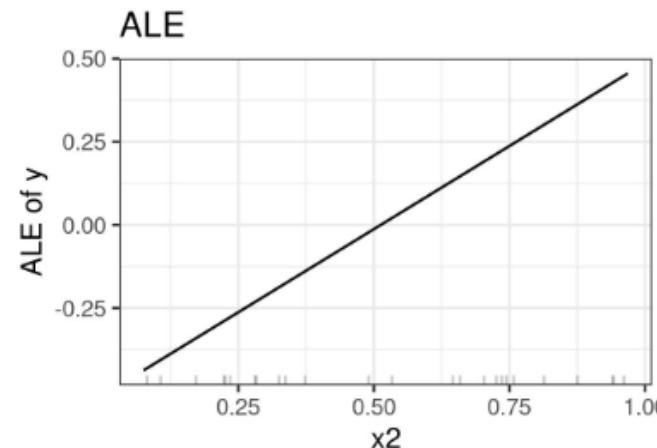
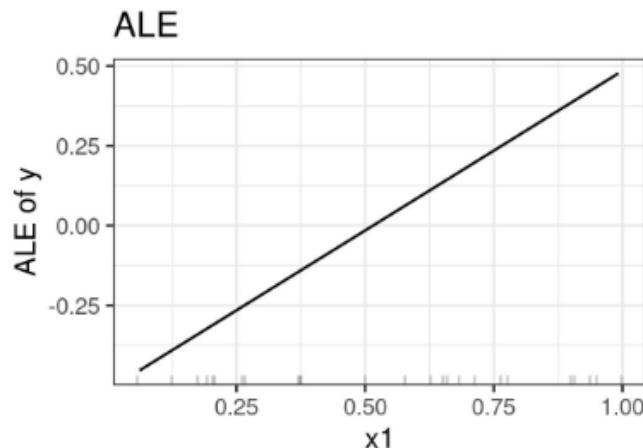
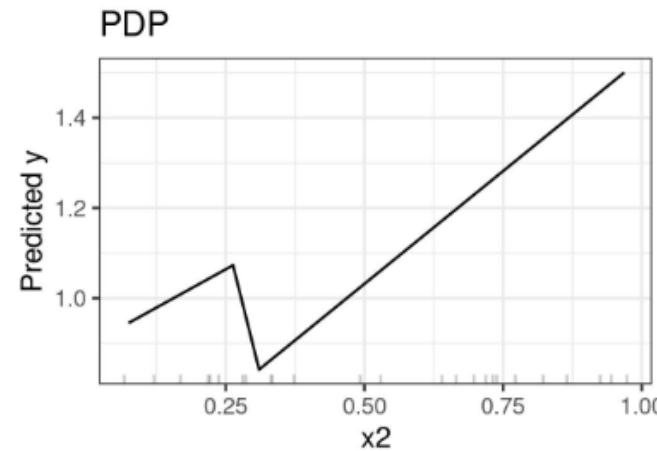
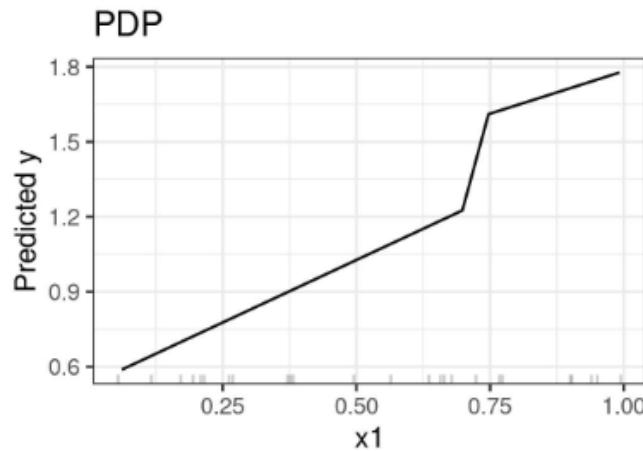
ALE Plot

- ▶ Examples: Two strongly correlated features (PDP fails here)
 - ▶ The model predicts the sum of the two features (shaded background), with the exception that if x_1 is greater than 0.7 and x_2 less than 0.3, the model always predicts 2.



ALE Plot

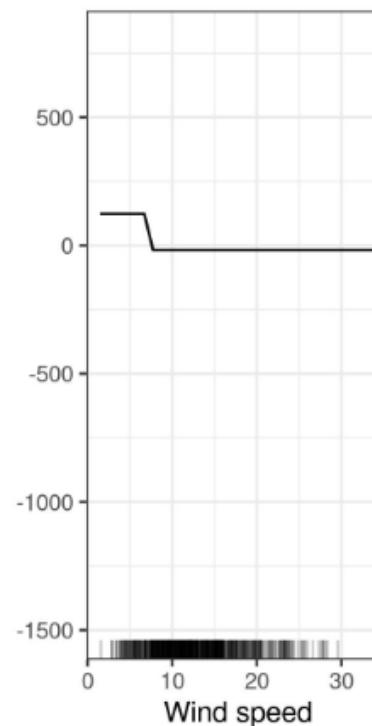
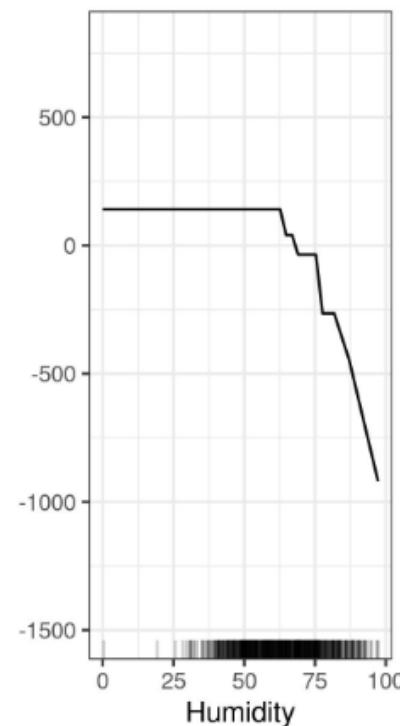
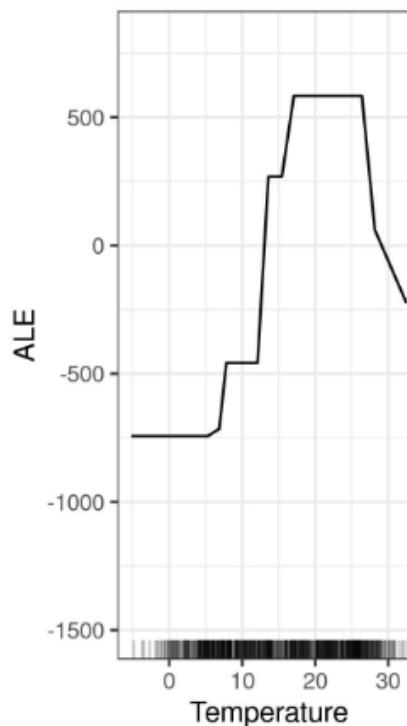
- ▶ Examples: Two strongly correlated features (PDP fails here)



ALE Plot

► Ex: Bike Rentals (model: regression tree)

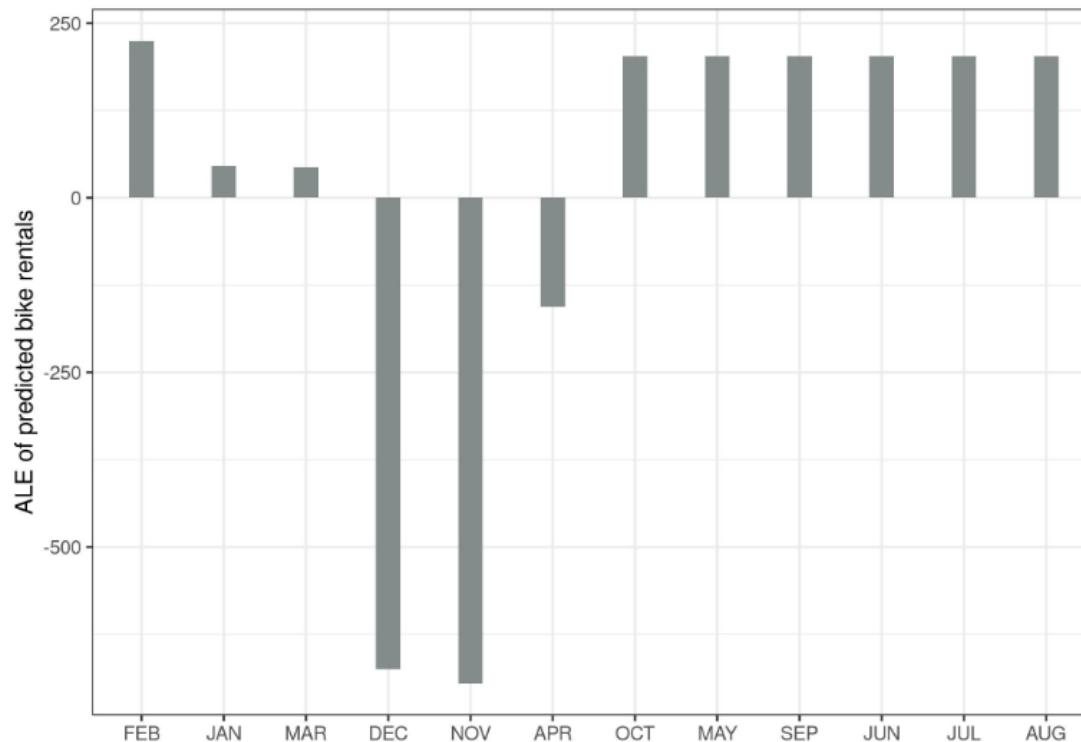
- ▶ Temp has a strong effect on the prediction: average prediction rises with increasing temperature but falls again above 25.
- ▶ Humidity has a negative effect: When above 60%, the higher the relative humidity, the lower the prediction.
- ▶ Wind speed does not affect the predictions much.



ALE Plot

► Ex: Bike Rentals (model: regression tree)

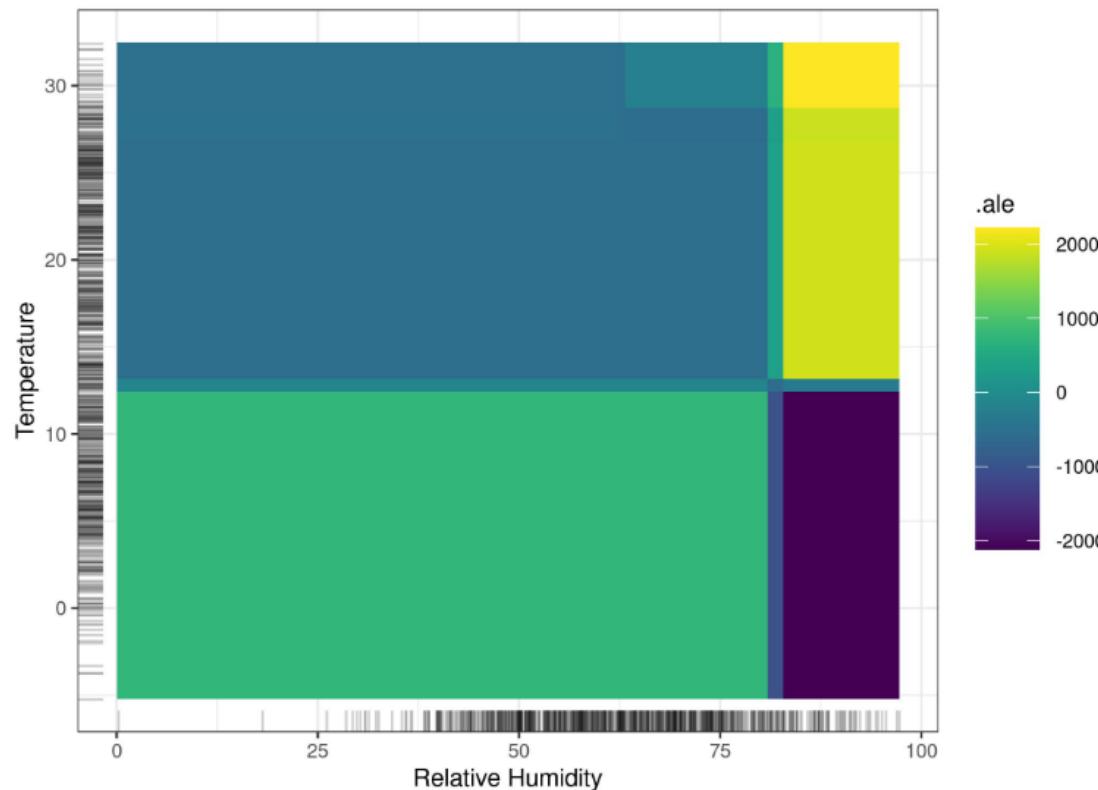
- ▶ The months are ordered by their similarity to each other, based on the distributions of the other features by month.
- ▶ December and November have a lower effect on the predicted number of rented bikes.



ALE Plot

► Ex: Bike Rentals (model: regression tree)

- ▶ Interaction between temp and humidity (second-order effect only, not the main)
- ▶ Hot and humid weather increases the prediction.
- ▶ In cold and humid weather an additional negative effect on the number of predicted bikes is shown.



ALE Plot

- ▶ Advantages:
 - ▶ Unbiased: works with correlated features
 - ▶ Faster to compute than PDPs: looks to the neighborhood only
 - ▶ Clear interpretation: ALE plots are centered at zero.
- ▶ Disadvantages:
 - ▶ Interpretation of the effect across intervals is not permissible
 - ▶ effects may differ from the coefficients specified in a linear regression model
 - ▶ E.g.: $\hat{f}(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2$ ALE will be nonlinear!!!
 - ▶ ALE plots can become a bit shaky with high intervals

Feature Interaction

- ▶ Interaction strength: How much of the variation of the prediction depends on the interaction of the features.
 - ▶ H-statistic (expensive to evaluate)
- ▶ Two versions:
 - ▶ Two-way interaction: Whether and to what extent **two** features in the model interact with each other

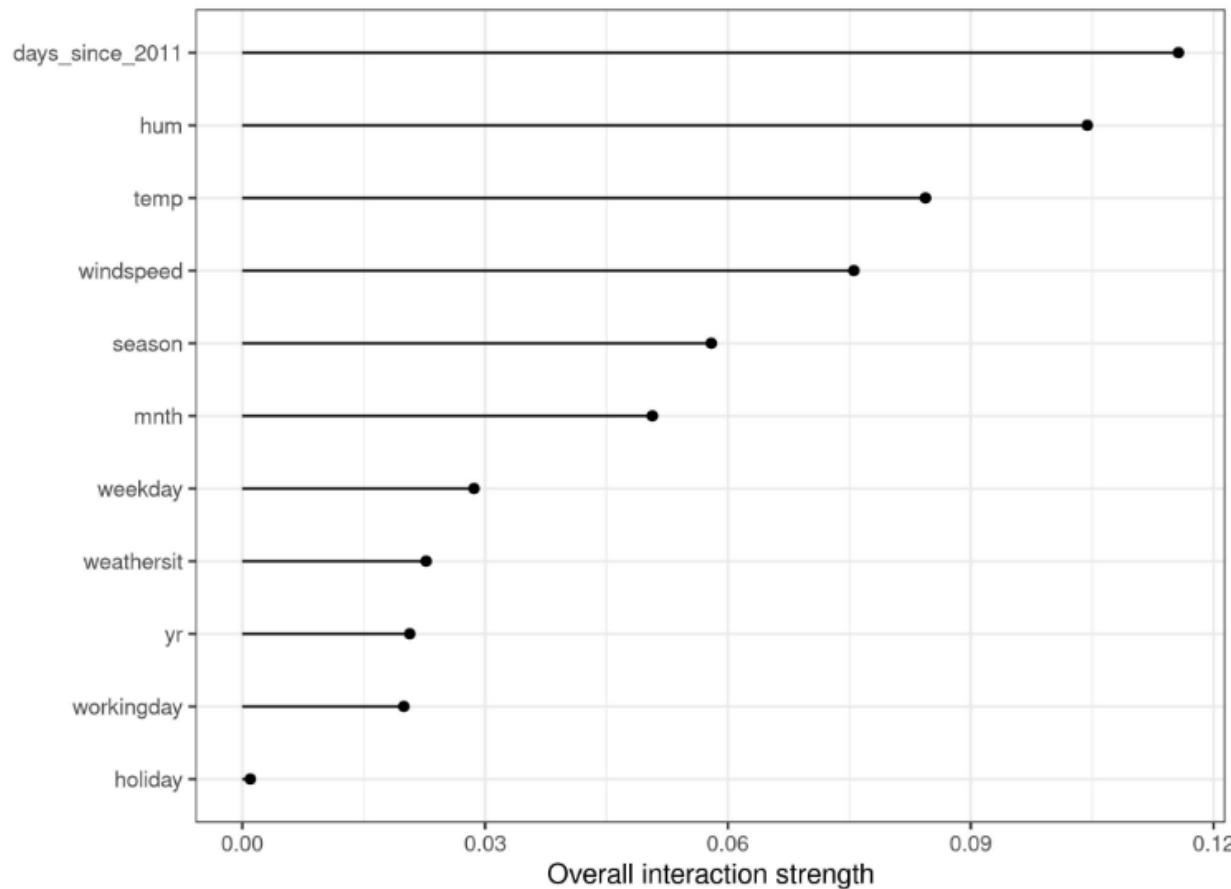
$$H_{jk}^2 = \frac{\sum_{i=1}^n \left[PD_{jk}(x_j^{(i)}, x_k^{(i)}) - PD_j(x_j^{(i)}) - PD_k(x_k^{(i)}) \right]^2}{\sum_{i=1}^n PD_{jk}^2(x_j^{(i)}, x_k^{(i)})}$$

- ▶ Total interaction: Whether and to what extent a feature interacts in the model with **all** the other features

$$H_j^2 = \frac{\sum_{i=1}^n \left[\hat{f}(x^{(i)}) - PD_j(x_j^{(i)}) - PD_{-j}(x_{-j}^{(i)}) \right]^2}{\sum_{i=1}^n \hat{f}^2(x^{(i)})}$$

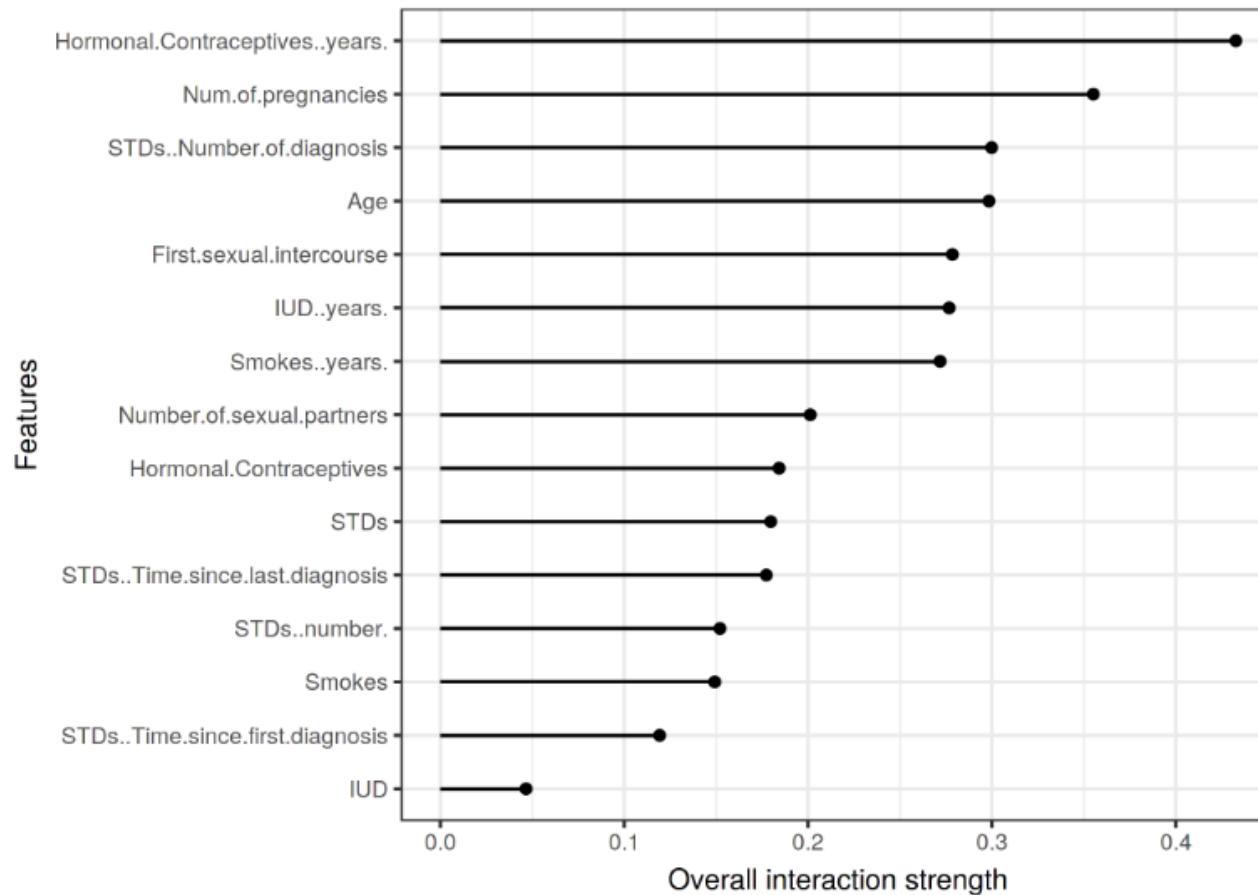
Feature Interaction

- ▶ Example: Bike Rental Dataset (trained model is SVM)
 - ▶ H-statistic: Interaction effects between the features are very weak (below 10% of variance explained per feature).



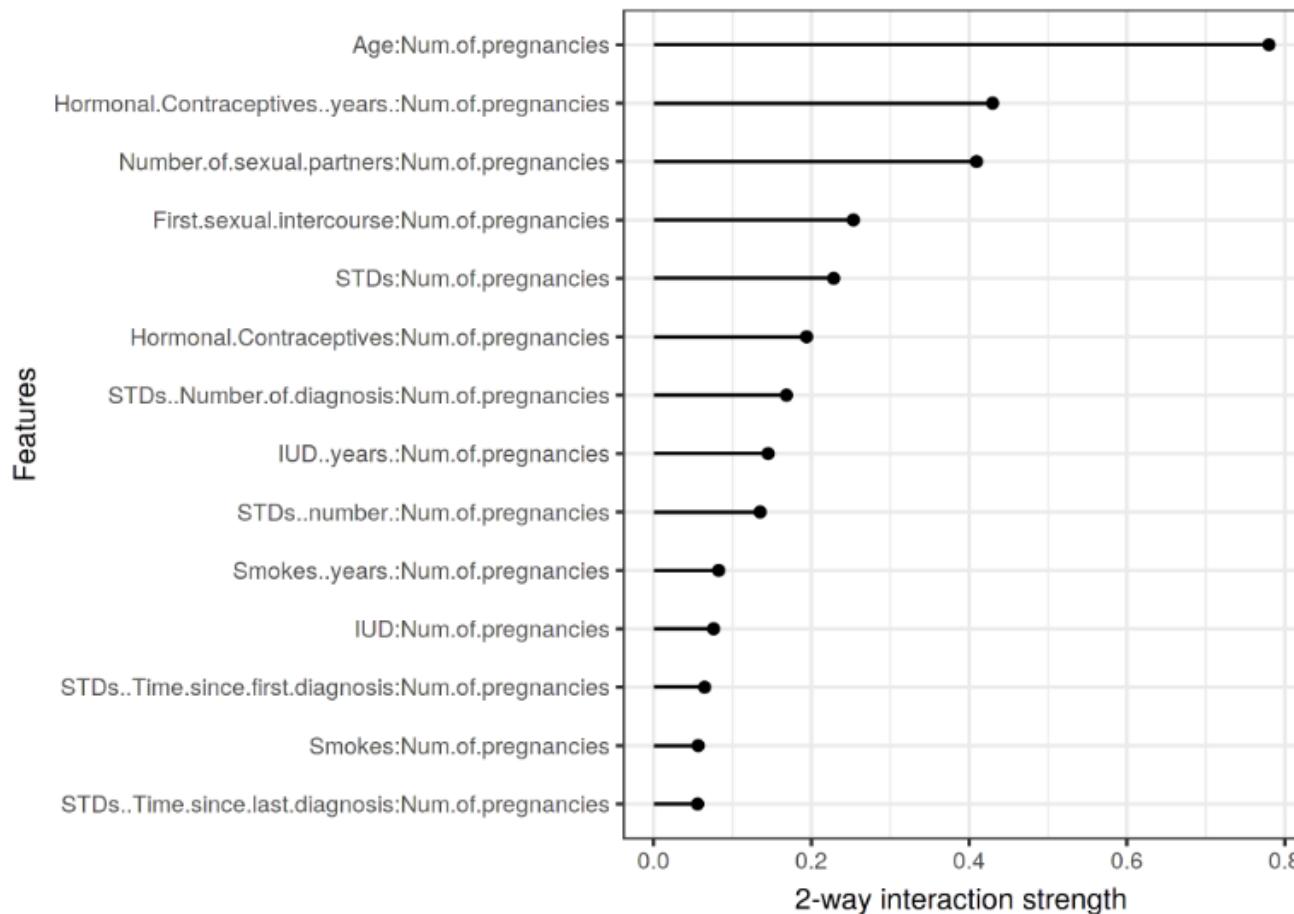
Feature Interaction

- ▶ Example: Cervical Cancer Dataset (trained model is RF)
 - ▶ H-statistic: The years on hormonal contraceptives has the highest relative interaction effect, then the number of pregnancies.



Feature Interaction

- ▶ Example: Cervical Cancer Dataset (trained model is RF)
 - ▶ Strong interaction between the number of pregnancies and the age



Feature Interaction

- ▶ Advantages
 - ▶ Meaningful Interpretation: The share of variance that is explained by the interaction
 - ▶ Dimensionless: Comparable across features and models
 - ▶ Detects all kinds of interactions
 - ▶ Possible to analyze arbitrary higher interactions

- ▶ Disadvantages
 - ▶ Computationally expensive
 - ▶ Results can be unstable (multiple runs)

Permutation Feature Importance

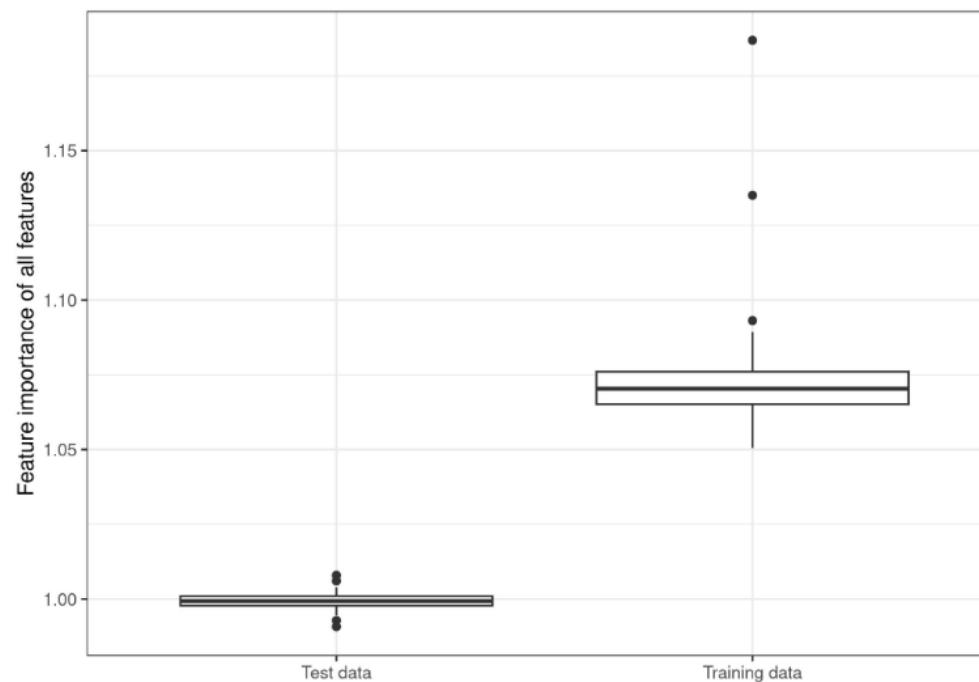
- ▶ Measures the increase in the prediction error of the model after permuting a feature's values
 - ▶ A feature is “important” if shuffling its values increases the model error
 - ▶ Introduced by Breiman (2001) for RF

Permutation Feature Importance

- ▶ The algorithm (Fisher, Rudin, Dominici 2018)
- ▶ Input: Trained model \hat{f} , feature matrix X, target vector y, error measure $L(y, \hat{f})$
 1. Estimate the original model error $e_{orig} = L(y, \hat{f}(X))$ (e.g., MSE)
 2. For each feature $j \in \{1, \dots, p\}$ do:
 1. Generate feature matrix X_{perm} by permuting feature j in the data X. This breaks the association between feature j and true outcome y.
 2. Estimate error $e_{perm} = L(y, \hat{f}(X_{perm}))$ based on the predictions of the permuted data.
 3. Calculate permutation feature importance as quotient $FI_j = e_{perm}/e_{orig}$ or difference $FI_j = e_{perm} - e_{orig}$
 3. Sort features by descending FI.

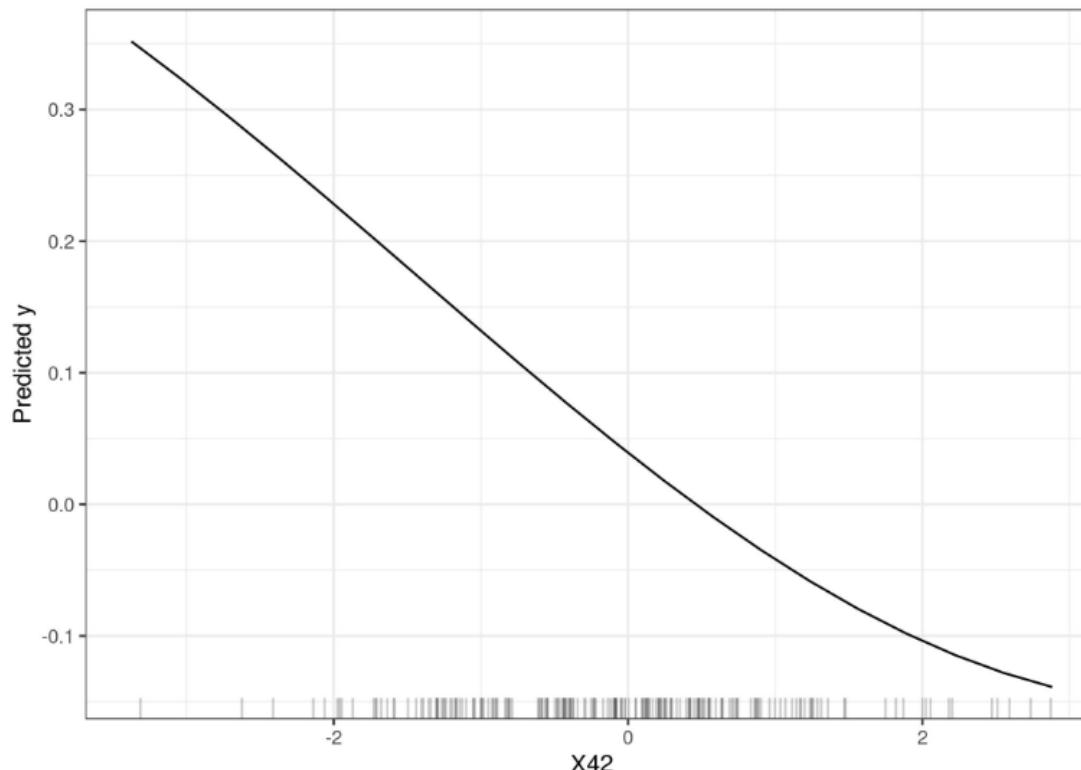
Permutation Feature Importance

- ▶ Should I use training data or test data?
 - ▶ The case for using test data: We are, after all, interested in making predictions for unseen data.
 - ▶ Ex: Let's generate a simulated dataset with 50 random features and an independent outcome variable of these features.
 - ▶ Results of SVM model: Training MAE: 0.29, Test MAE: 0.82



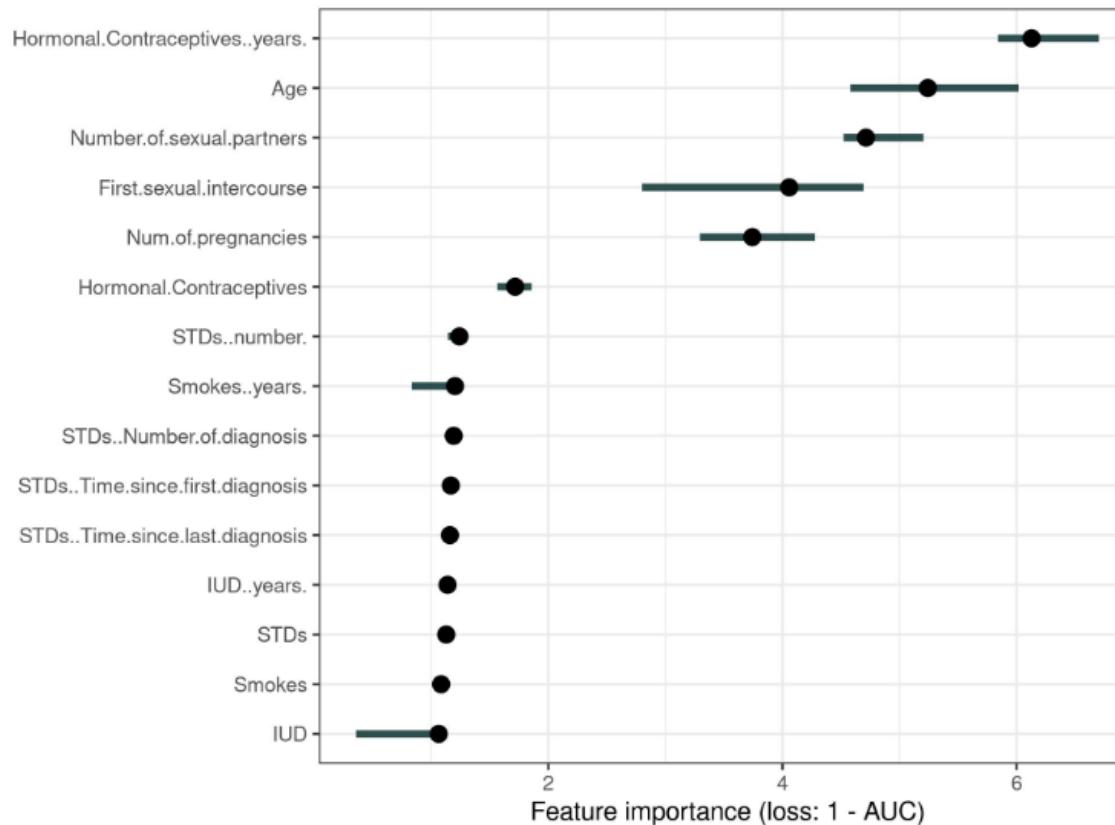
Permutation Feature Importance

- ▶ Should I use training data or test data?
 - ▶ The case for using training data: We are, after all, interested in understanding how the model behaves.
 - ▶ Ex: Same as before. Let's look a PDP for X42. SVM has learned to rely on feature X42 for its predictions.



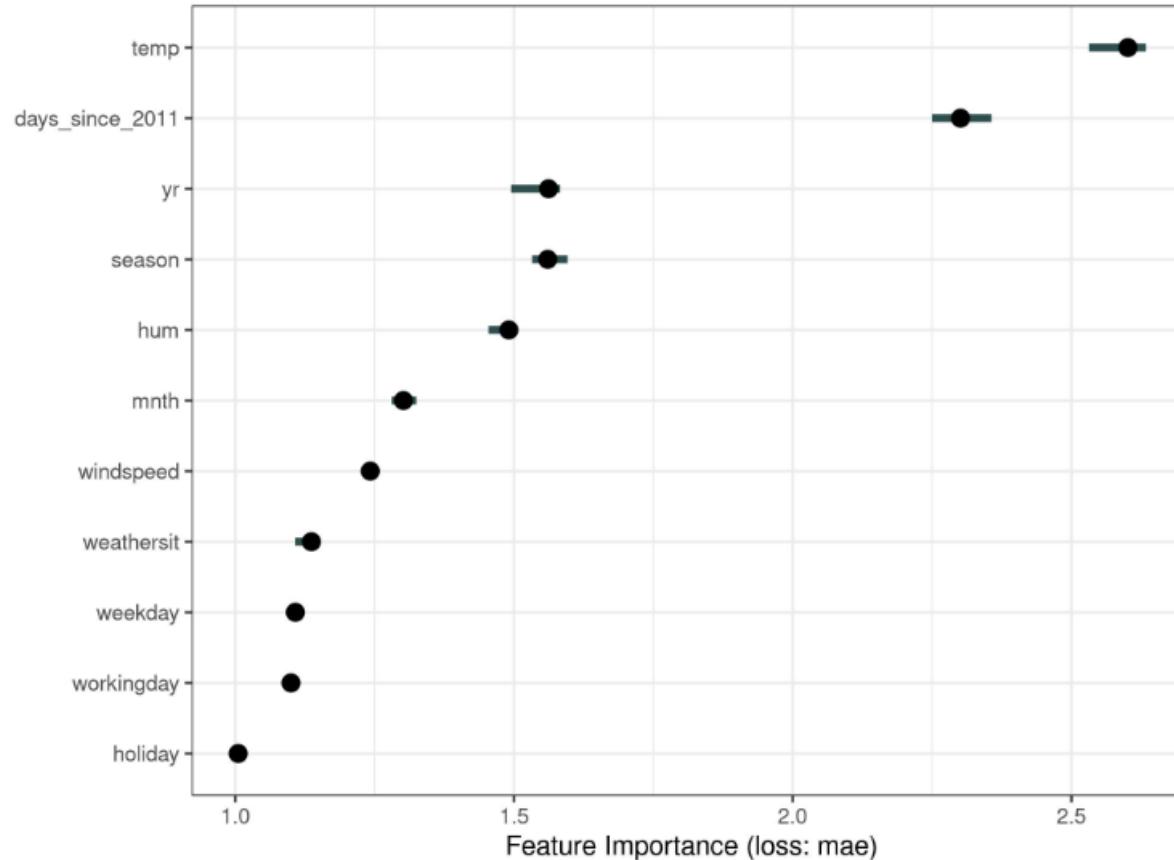
Permutation Feature Importance

- ▶ Ex: Cervical Cancer Dataset
 - ▶ Training a RF Model
 - ▶ Permuting Hormonal.Contraceptives..years. resulted in an increase in 1-AUC by a factor of 6.13



Permutation Feature Importance

- ▶ Ex: Bike Sharing Dataset
 - ▶ Training a SVM Model



Permutation Feature Importance

▶ Advantages

- ▶ Nice interpretation: increase in model error when the feature's information is destroyed.
- ▶ Comparable across different problems
- ▶ Automatically considers all interactions
- ▶ No need to retrain the model

▶ Disadvantages

- ▶ FI is linked to the error of the model
- ▶ Randomness in shuffling: Results might vary greatly
- ▶ If features are correlated, the permutation feature importance can be biased by unrealistic data instances.
- ▶ Adding a correlated feature can decrease the importance of the associated feature by splitting the importance between both features.

Global Surrogate Models

- ▶ Global surrogate model: An interpretable model trained to approximate the predictions of a black box model.
 - ▶ We can draw conclusions about the black box model by interpreting the surrogate model.
- ▶ Goals:
 - ▶ (1) To approximate the predictions of the black-box model as accurately as possible,
 - ▶ (2) To be interpretable.
- ▶ Could use any of the interpretable models discussed earlier
- ▶ Training a surrogate model is model-agnostic: No info needed about how the model works, just need data and the prediction function of the model.

Global Surrogate Models

- ▶ Steps:
 1. Select a dataset X (could be training set or a new dataset or subset)
 2. For the selected dataset X , get the predictions of the black box model.
 3. Select an interpretable model type.
 4. Train the interpretable model on the dataset X and its predictions.
 5. Congratulations! You now have a surrogate model.
 6. Measure how well the surrogate model replicates the predictions of the black box model.
 7. Interpret the surrogate model.

Global Surrogate Models

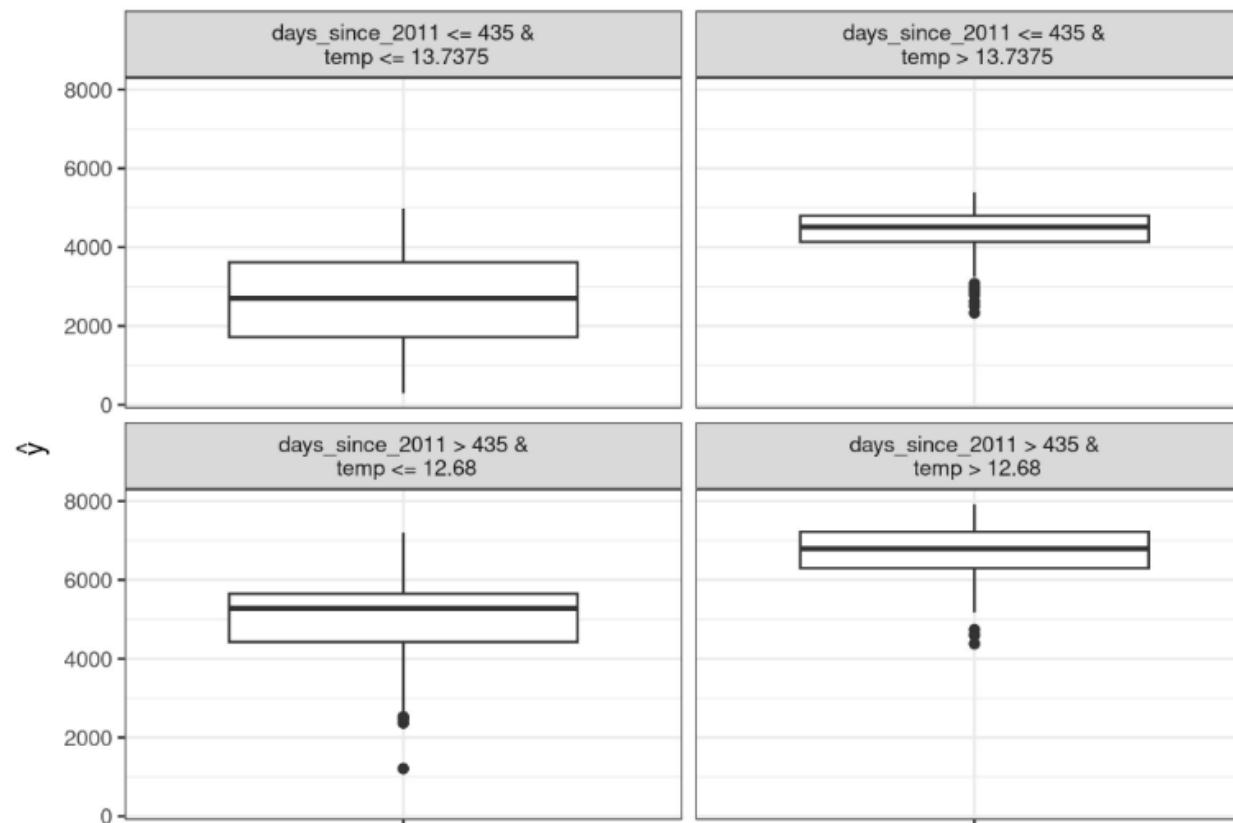
- ▶ How to measure the performance of a surrogate model:

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^n (\hat{y}_*^{(i)} - \hat{y}^{(i)})^2}{\sum_{i=1}^n (\hat{y}^{(i)} - \bar{\hat{y}})^2}$$

- ▶ $\hat{y}_*^{(i)}$: prediction for the i-th datapoint of the surrogate model
- ▶ $\hat{y}^{(i)}$: prediction for the i-th datapoint of the black-box model
- ▶ $\bar{\hat{y}}$: mean black-box prediction
- ▶ R-squared measures the percentage of variance that is captured by the surrogate model.
- ▶ R-squared is close to 1 implies the surrogate model approximates the behavior of the black box model very well.
- ▶ Independent from the prediction accuracy of the black-box model.

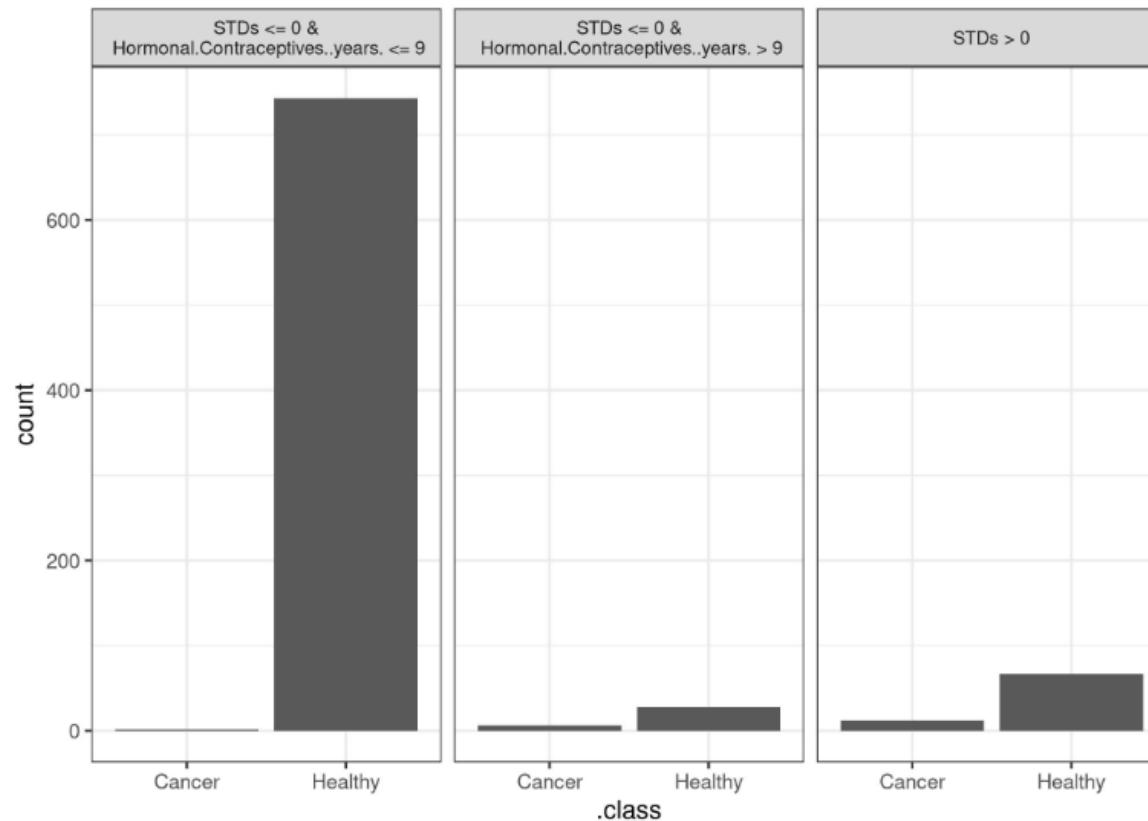
Global Surrogate Models

- ▶ Example: Bike Rentals Dataset (SVM model)
 - ▶ Let's use a decision tree (DT) as a surrogate model
 - ▶ Surrogate model has 0.77 R-squared value. What does it mean?
 - ▶ The following is box-plots for the predictions of 4 leaf nodes using the DT



Global Surrogate Models

- ▶ Ex: Cervical Cancer Dataset (RF model)
 - ▶ Let's use a decision tree (DT) as a surrogate model
 - ▶ Surrogate model has 0.19 R-squared value. What does it mean?
 - ▶ The following is box-plots for the predictions of 4 leaf nodes using the DT



Global Surrogate Models

- ▶ Advantages
 - ▶ Flexible: Any interpretable model could be used as surrogate
 - ▶ Intuitive and straightforward
 - ▶ R-squared can measure performance of the surrogate model.

- ▶ Disadvantages
 - ▶ Beware: Conclusions are about the model and not the data!!!
 - ▶ Surrogate model fit could be good for some subset of the data and not good for some other subset.

ÖZYEĞİN ÜNİVERSİTESİ

DS 530

Fairness and Interpretability

ENİS KAYIŞ

Local Model-Agnostic Methods

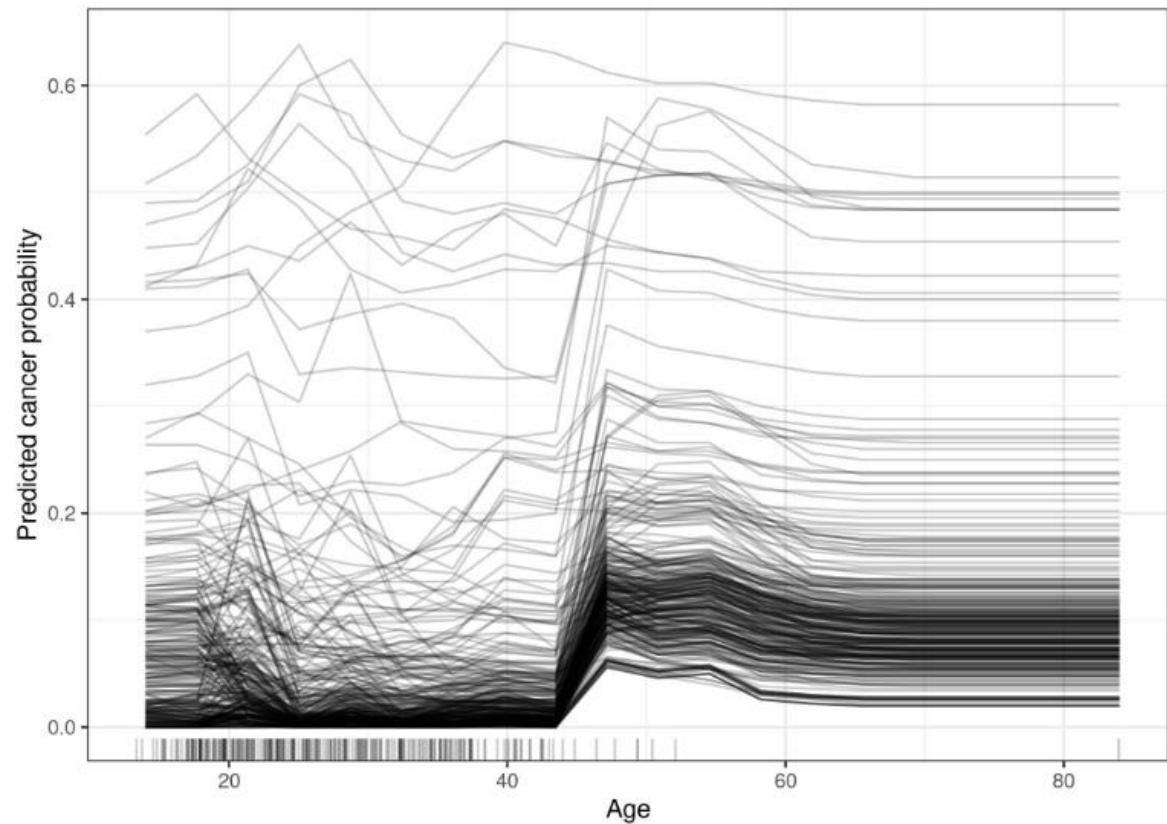
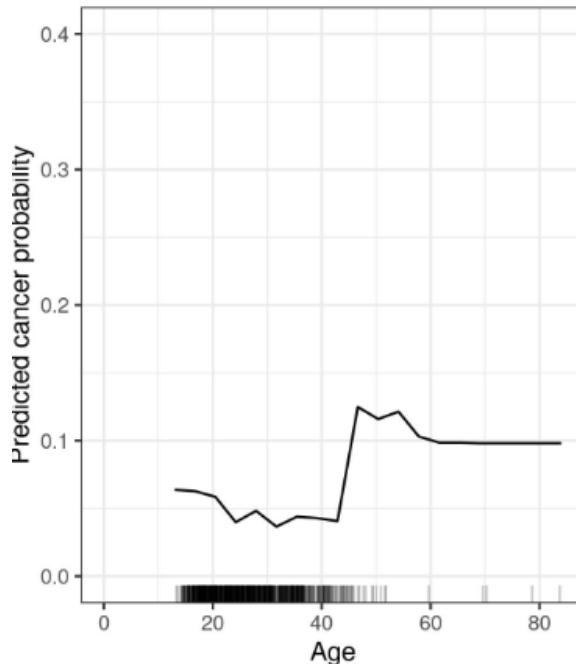
- ▶ Explain individual predictions.
 - ▶ **Individual conditional expectation curves:** Describe how changing a feature changes the prediction.
 - ▶ **Local surrogate models (LIME):** Explain a prediction by replacing the complex model with a locally interpretable surrogate model
 - ▶ **Scoped rules (anchors):** Rules to describe which feature values anchor a prediction, in the sense that they lock the prediction.
 - ▶ **Counterfactual explanations:** Explain a prediction by finding which features would need to be changed to achieve a desired prediction.
 - ▶ **Shapley values:** Fairly assigns the prediction to individual features.
 - ▶ **SHAP:** Alternative method for Shapley values.

Individual Conditional Expectation (ICE)

- ▶ Shows how the instance's prediction changes when a feature changes
 - ▶ PDP: The average effect of a feature. It is a global method (not focus on specific instances).
 - ▶ The equivalent to a PDP for individual data instances is ICE plot
- ▶ ICE vs PDPs:
 - ▶ PDPs can obscure a heterogeneous relationship created by interactions.
 - ▶ In case of interactions, the ICE plot will provide more insight.

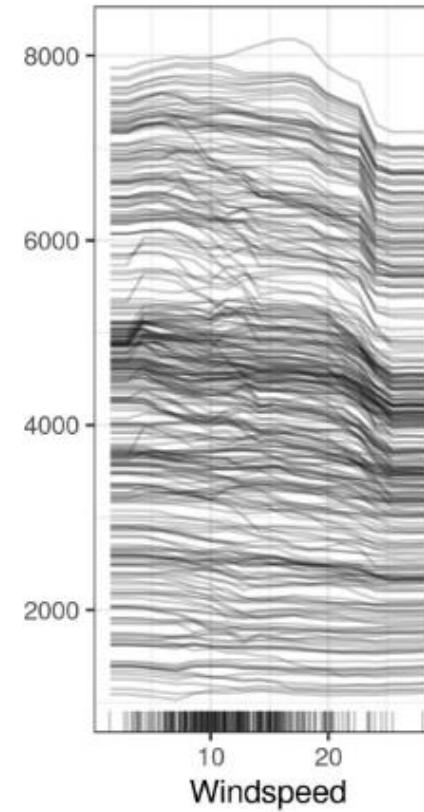
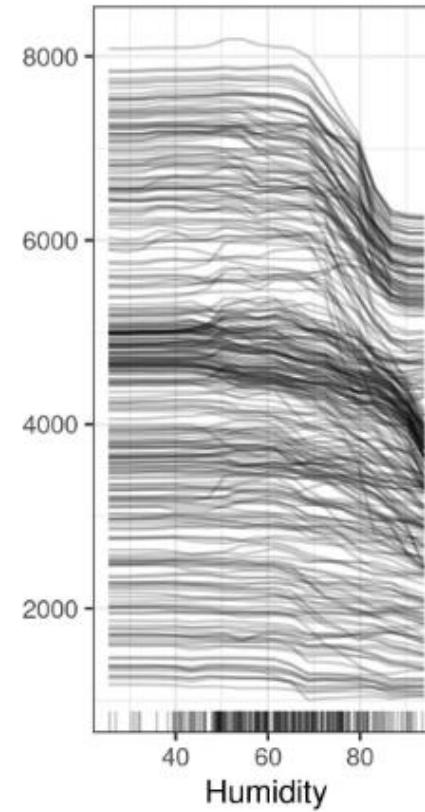
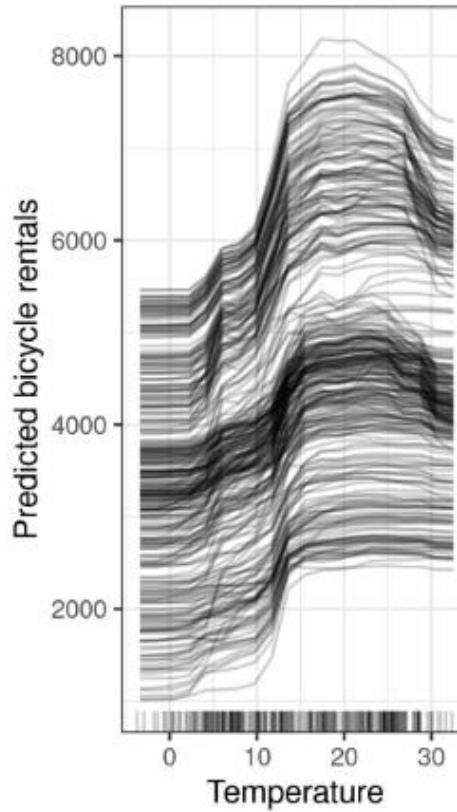
Individual Conditional Expectation (ICE)

- ▶ Example: Cervical Cancer Dataset (RF model)
 - ▶ PDP: probability increases around the age of 50
 - ▶ ICE: Similar pattern for most women, but for the few women with a high predicted probability at a young age, the probability does not change much with age.



Individual Conditional Expectation (ICE)

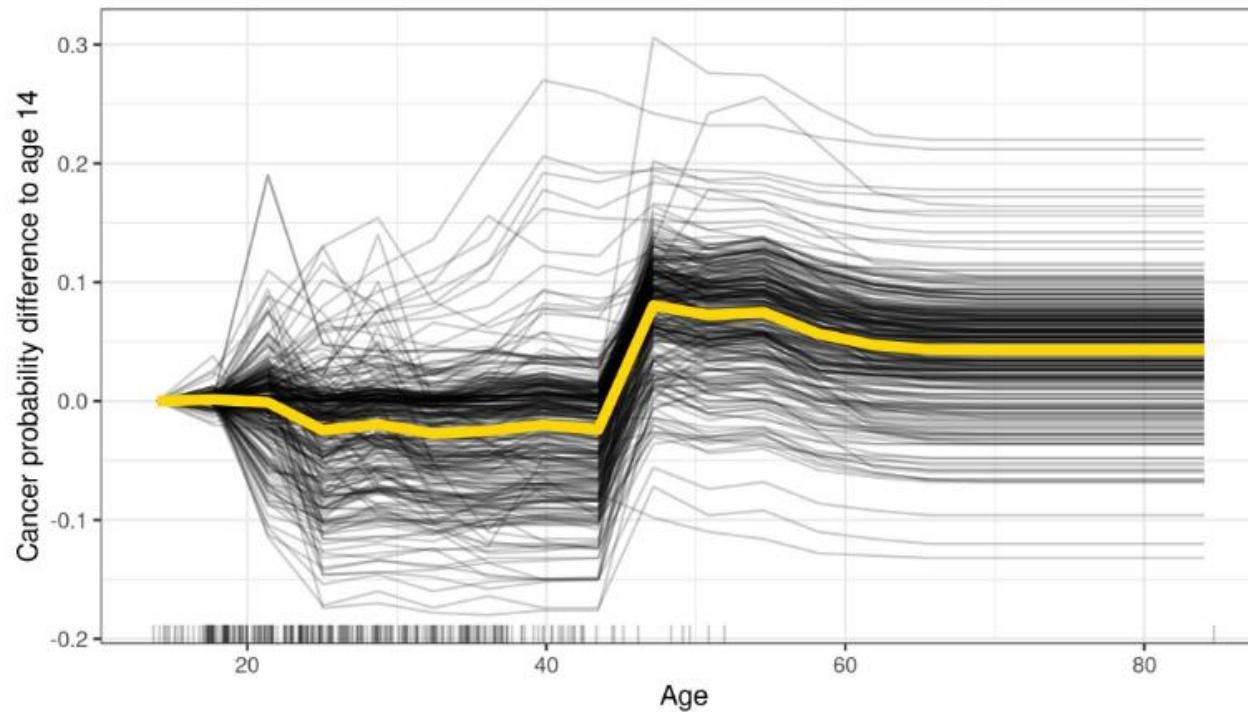
- ▶ Ex: Bike Rentals Dataset (RF model)
 - ▶ All curves are similar, means no obvious interactions.
 - ▶ PDP is a good summary of the relationships between the features and the predicted number of bicycles



Individual Conditional Expectation (ICE)

► Centered ICE plot (c-ICE)

- Sometimes it is hard to tell whether the ICE curves differ between individuals as they start at different predictions.
- Solution: Center the curves at a certain point in the feature.
- Ex: c-ICE plot for predicted cancer probability by age. Lines are fixed to 0 at age 14.



Individual Conditional Expectation (ICE)

- ▶ Advantages:
 - ▶ Even more intuitive to understand than PDPs
 - ▶ Unlike PDPs, can uncover heterogeneous relationships.
- ▶ Disadvantages:
 - ▶ Can only display one feature meaningfully
 - ▶ If features are correlated, then some points in the lines might be invalid data points
 - ▶ Plot can become overcrowded if too many datapoints
 - ▶ Might not be easy to see the average

Announcement

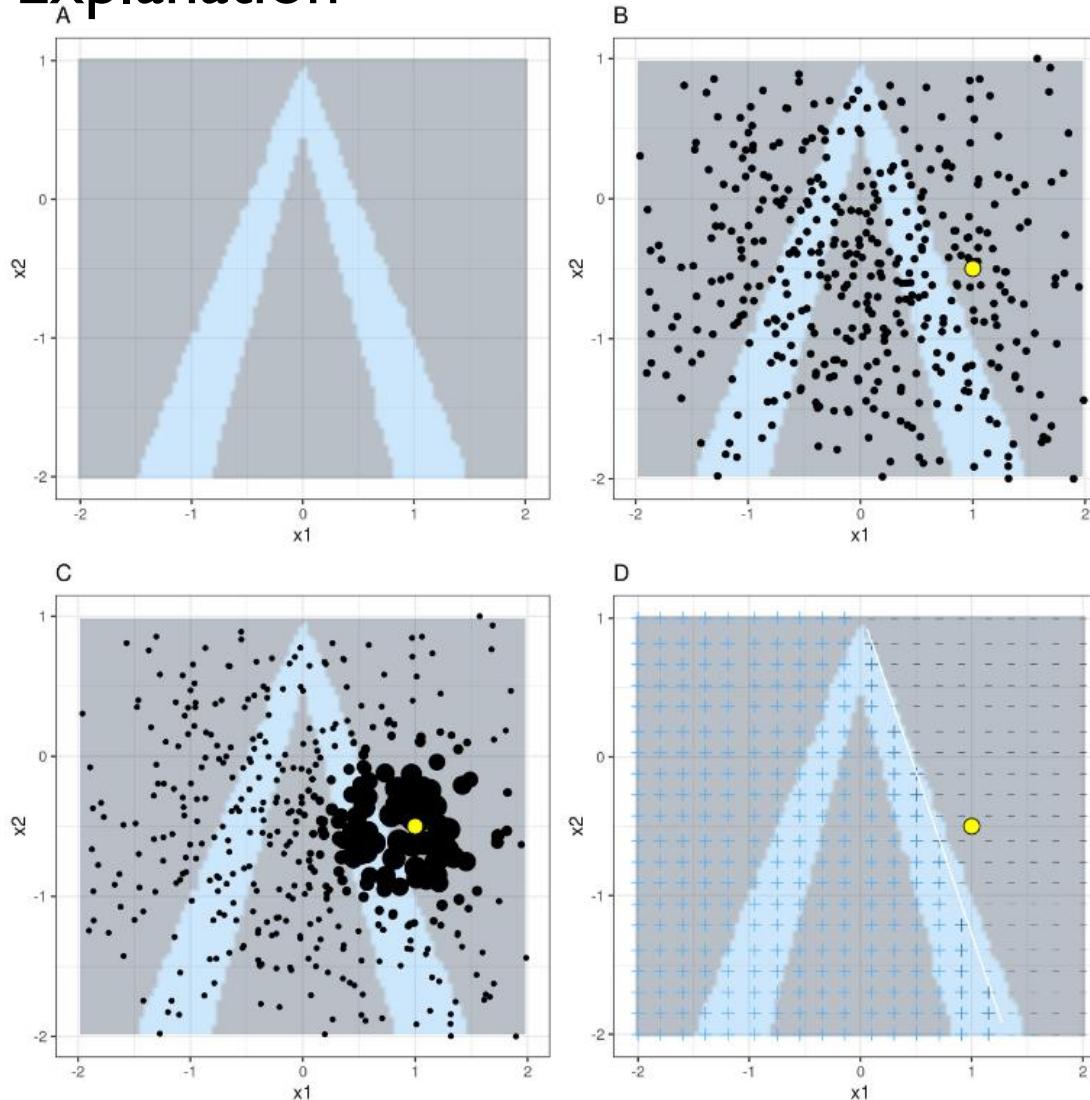
- ▶ Midterm I is next week (18 April)
- ▶ All subjects covered by the end of this class are included
- ▶ Two parts (50% each)
 - ▶ Part I (Written Exam)
 - ▶ Time: 17:45-19:15
 - ▶ Open Books and Notes, but no electrical device except calculators
 - ▶ Requires understanding of the subjects at a conceptual level
 - ▶ Part 2 (Take Home Exam)
 - ▶ Time: April 18 19:30- April 20 19:30
 - ▶ Requires writing codes for making interpretations and discussion of the results

Local Surrogate (LIME)

- ▶ Local interpretable model-agnostic explanations (LIME)
 - ▶ Interpretable models are used to explain individual predictions of black box models
- ▶ Idea:
 1. Train your black-box model using your dataset and get the prediction function.
 2. Generate a new dataset consisting of perturbed samples and their predictions using the black box model (test what happens to the predictions when data varies based on your model)
 3. On this new dataset, train an interpretable model, which is weighted by the proximity of the sampled instances to the instance of interest.
- ▶ The learned model should be a good approximation of the black-box model predictions locally, but it does not have to be a good global approximation.

Local Surrogate (LIME)

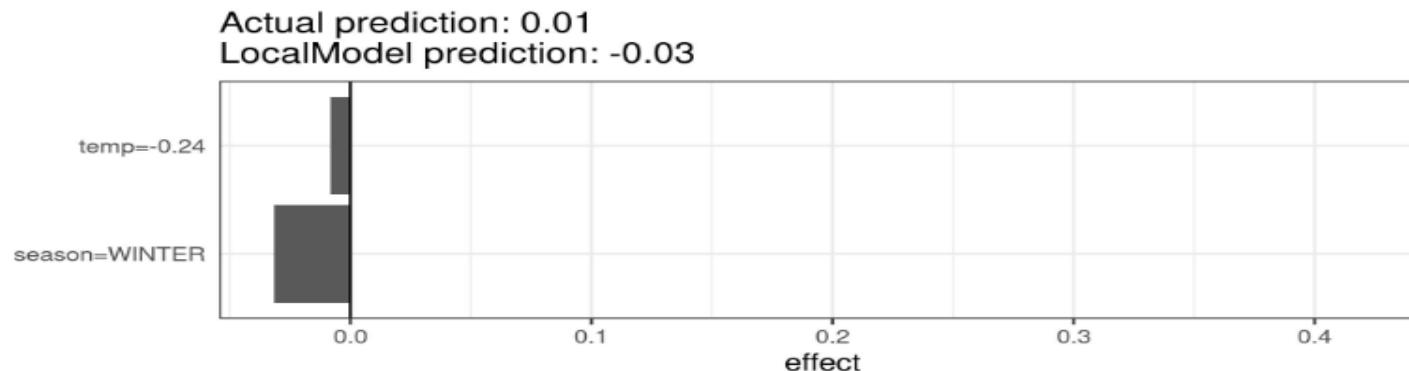
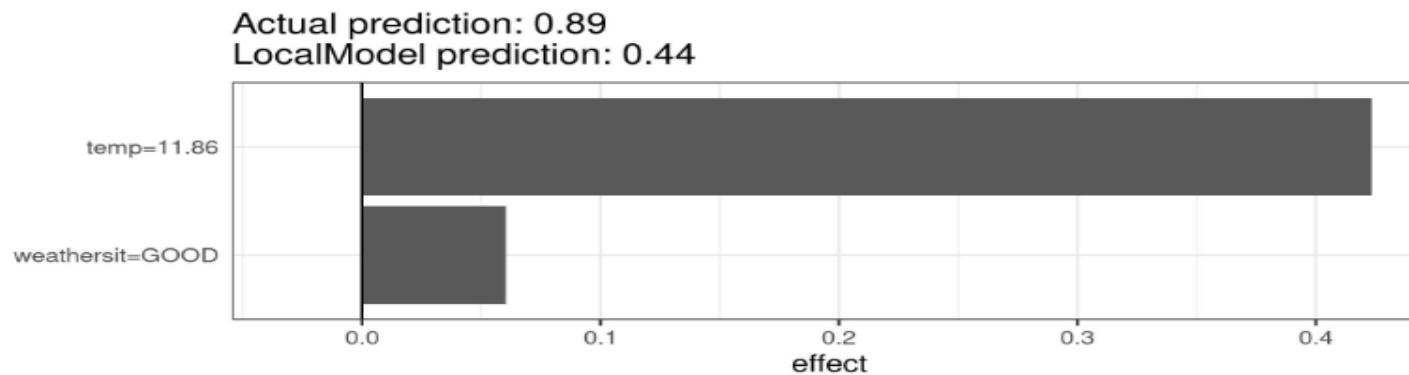
► Pictorial Explanation



Local Surrogate (LIME)

► Ex: Bike Rental Dataset

- ▶ Classification problem: Whether the number of bicycles rented on a day will be above or below the trend line (i.e., adjust for trend)
- ▶ RF with 100 trees, then sparse local linear model as surrogate
- ▶ x-axis shows the feature effect



Local Surrogate (LIME)

- ▶ Advantages:
 - ▶ Flexibility wrt the black-box model used
 - ▶ Explanations are short (i.e., selective) and possibly contrastive if variables are few or DT is shallow
 - ▶ Good in applications where the recipient of the explanation is a lay person.
 - ▶ Not so good for complete attributions (legally required to fully explain a prediction or debugging).
 - ▶ Local surrogate models can use other (interpretable) features than the original model was trained on
- ▶ Disadvantages:
 - ▶ Sensitive to the definition of neighborhood
 - ▶ Instability of the explanations
 - ▶ Explanations can be manipulated by the data scientist to hide biases

Counterfactual Explanations

- ▶ Describes a causal situation in the form: “If X had not occurred, Y would not have occurred”
 - ▶ Ex: “If I hadn’t taken a sip of this hot coffee, I wouldn’t have burned my tongue”.
 - ▶ Counterfactual: Consider the case where I haven’t drunk hot coffee
- ▶ Event: predicted outcome of an instance
- ▶ Causes: particular feature values of this instance that “caused” a certain prediction.
- ▶ Beware: Correlation does not imply causation, yet here we are simply talking about the prediction, not the actual process.

Counterfactual Explanations

- ▶ Idea:
 - ▶ Change the feature values of an instance , then analyze how the prediction changes.
 - ▶ Interested in scenarios in which the prediction changes in a relevant way, like a flip in predicted class
 - ▶ A counterfactual explanation describes the **smallest change to the feature values** that changes the prediction
- ▶ Use Case:
 - ▶ Ex:What is the rental price of a house?
 - ▶ Assume that a ML model predicts a rental price of 900 EUR.
 - ▶ What should change so that I can charge 1000 EUR or more?
 - ▶ If the size were 15 m² larger. Interesting, but nothing can be done.
 - ▶ If pets are allowed and better windows are installed, price would be 1000 EUR.

Counterfactual Explanations

- ▶ Qualities of a good counterfactual explanation:
 - ▶ A counterfactual instance produces the predefined prediction as closely as possible (not always possible e.g., imbalanced data)
 - ▶ A counterfactual should be as similar as possible to the instance regarding feature values
 - ▶ A counterfactual should change as few features as possible
 - ▶ A counterfactual instance should have feature values that are likely
 - ▶ Generate multiple diverse counterfactual explanations

Counterfactual Explanations

- ▶ How to generate counterfactuals?
 - ▶ Naïve way: Randomly change feature values of the instance of interest and stop when the desired output is predicted
 - ▶ Better Way:
 - ▶ Define a loss function based on the criteria mentioned before.
 - ▶ Find the counterfactual explanation that minimizes this loss using an optimization algorithm.
- ▶ Dandl et al. (2020)
 - ▶ Loss function

$$L(x, x', y', X^{obs}) = \left(o_1(\hat{f}(x'), y'), o_2(x, x'), o_3(x, x'), o_4(x', X^{obs}) \right)$$

The diagram illustrates the components of the loss function L defined above. Four colored brackets under the terms o_1 , o_2 , o_3 , and o_4 are connected by arrows to text labels at the bottom:

- A blue bracket under o_1 points to the text "Distance between prediction and desired outcome".
- A red bracket under o_2 points to the text "Distance between instance and the counterfactual instance".
- A green bracket under o_3 points to the text "Sparse feature changes".
- An orange bracket under o_4 points to the text "Distance between dataset and counterfactual instance".

Counterfactual Explanations

- ▶ Example: German Credit Risk dataset (522 observations and 9 features)
 - ▶ Trained a SVM to predict the probability that a customer has a good credit risk
 - ▶ Goal: Find counterfactual for the following customer. SVM predicts good risk with 24.2%. What should change so that probability is larger than 50%?

age	sex	job	housing	savings	amount	duration	purpose
58	f	unskilled	free	little	6143	48	car

Counterfactual Explanations

- ▶ Example: German Credit Risk dataset (522 observations and 9 features)
- ▶ Ten best counterfactuals (none is dominated)

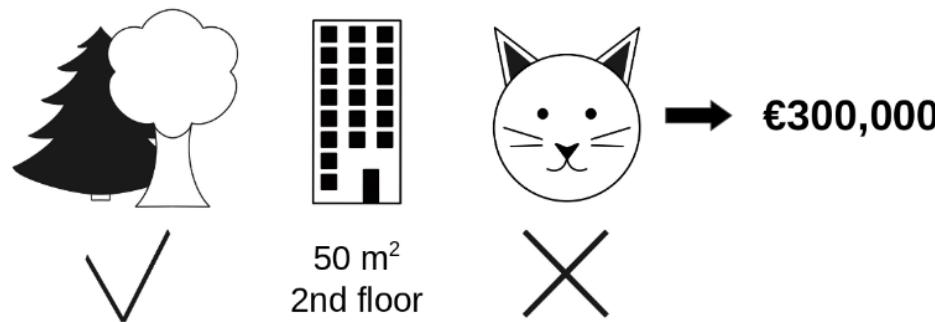
age	sex	job	amount	duration	o_2	o_3	o_4	$\hat{f}(x')$
		skilled		-20	0.108	2	0.036	0.501
		skilled		-24	0.114	2	0.029	0.525
		skilled		-22	0.111	2	0.033	0.513
-6		skilled		-24	0.126	3	0.018	0.505
-3		skilled		-24	0.120	3	0.024	0.515
-1		skilled		-24	0.116	3	0.027	0.522
-3	m			-24	0.195	3	0.012	0.501
-6	m			-25	0.202	3	0.011	0.501
-30	m	skilled		-24	0.285	4	0.005	0.590
-4	m		-1254	-24	0.204	4	0.002	0.506

Counterfactual Explanations

- ▶ Advantages:
 - ▶ The interpretation of counterfactual explanations is very clear.
 - ▶ Counterfactual method does not require access to the data or the model. (no need to reveal the model)
 - ▶ Works also with systems that do not use machine learning.
 - ▶ Relatively easy to implement
- ▶ Disadvantages:
 - ▶ Multiple counterfactual explanations (Rashomon effect)
 - ▶ One counterfactual might say to change feature A, the other counterfactual might say to leave A the same but change feature B, which is a contradiction.

Shapley Values

- ▶ Shapley values: how to **fairly** distribute the prediction value among the features.
- ▶ Ex: Trained a ML model to predict house price



- ▶ The average prediction for all apartments is €310,000.
- ▶ How much has each feature value contributed to the prediction compared to the average prediction?
- ▶ Clearly very simple using interpretable models. But what about black-box models?

Shapley Values

- ▶ Ex: park-nearby, cat-banned, area-50 and floor-2nd worked together to arrive at 300K prediction.
 - ▶ Why is there a difference of -10K?
 - ▶ Ex: 30K from park-nearby, 10K from area-50, 0 from floor-2nd, -50K from cat-banned.
- ▶ Shapley value: average marginal contribution of a feature value across all possible coalitions
 - ▶ Ex: What is the value of cat-banned? Many coalitions are possible
 - ▶ Consider park-nearby and area-50 combination



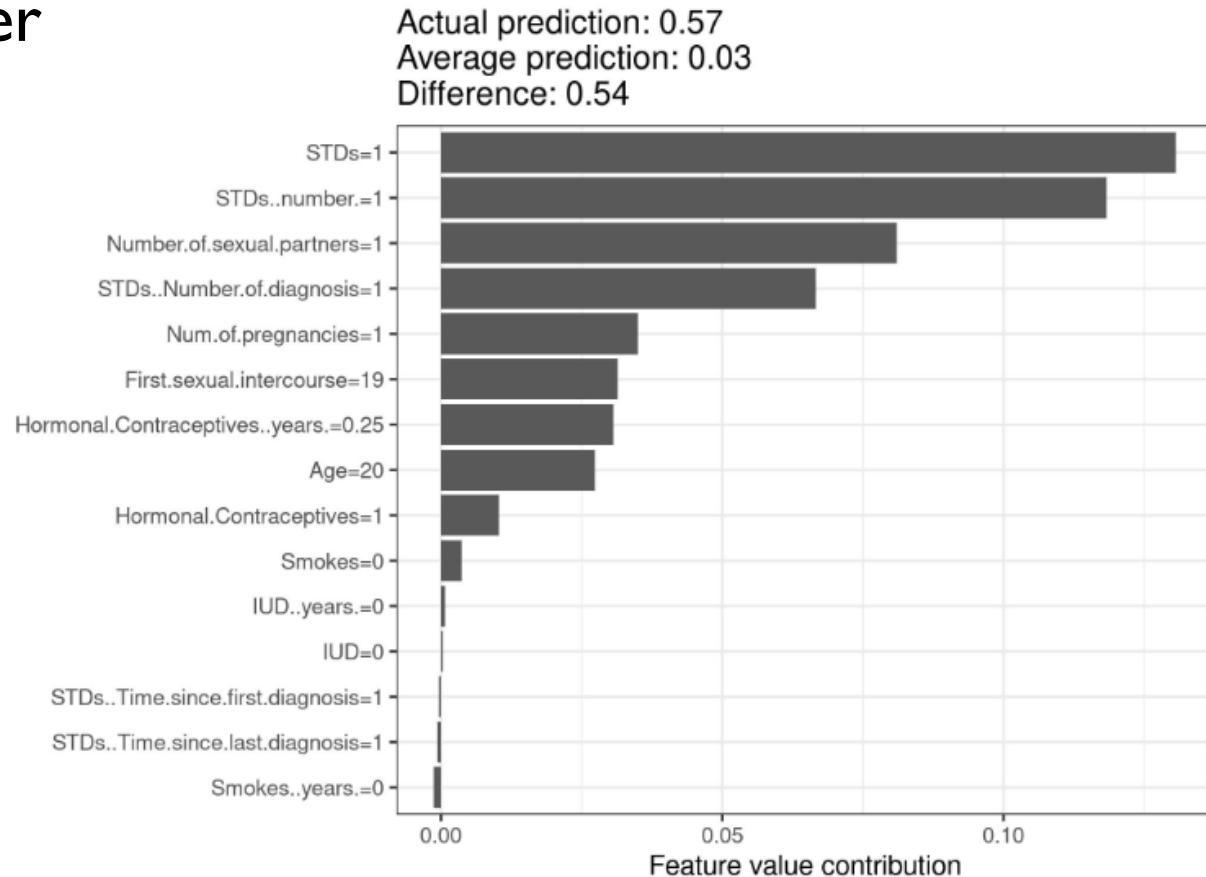
Shapley Values

- ▶ Repeat this computation for all possible coalitions.
- ▶ Shapley value is the average of all the marginal contributions to **all possible coalitions**.
 - No feature values
 - park-nearby
 - area-50
 - floor-2nd
 - park-nearby + area-50
 - park-nearby + floor-2nd
 - area-50 + floor-2nd
 - park-nearby + area-50 + floor-2nd .
- ▶ The computation time increases exponentially with the number of features.

Shapley Values

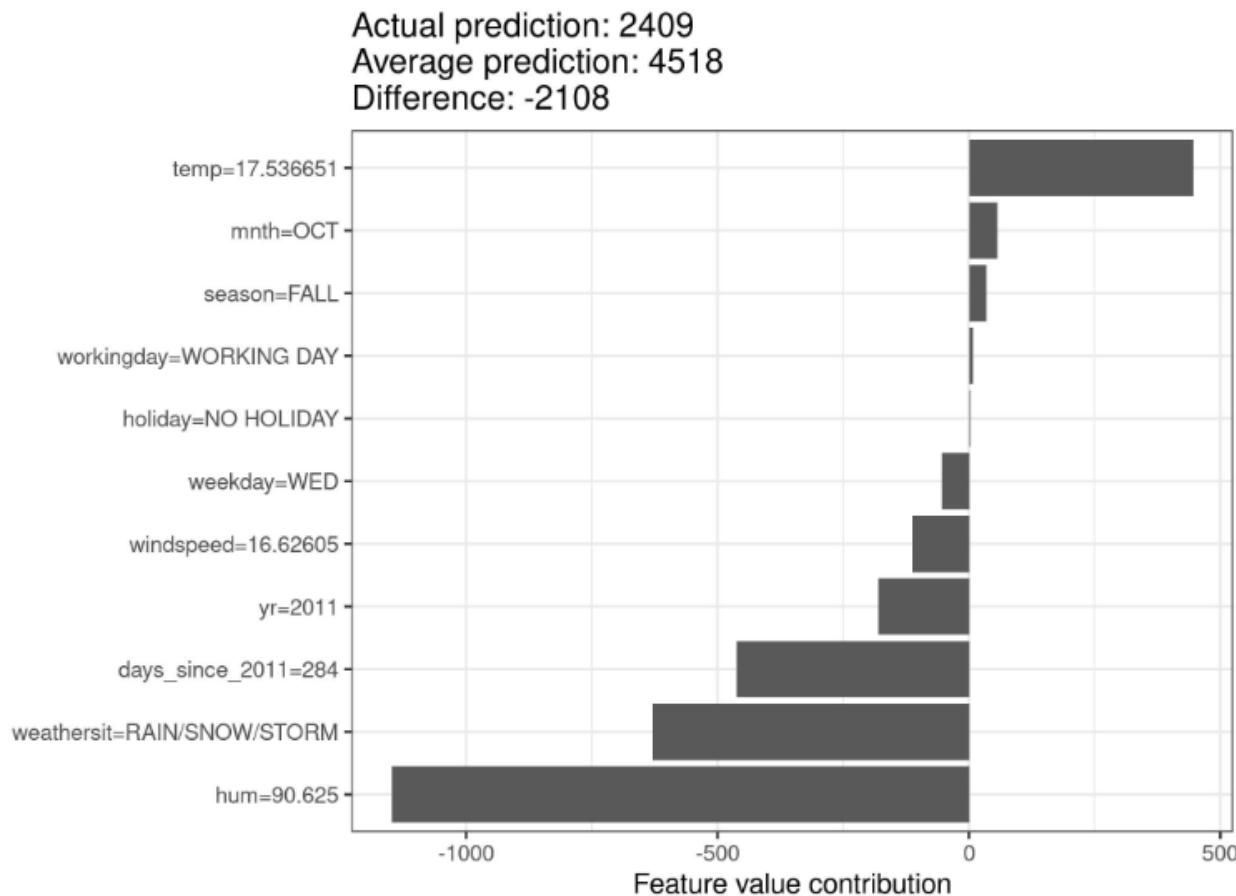
- ▶ Interpretation: How much a feature contributed to the prediction of this particular instance compared to the average prediction for the dataset?
- ▶ Ex: Cervical Cancer

- The actual prediction: 0.57
- The existence of STDs increased the probability the most.
- The sum of contributions yields the difference between actual and average prediction (0.54).



Shapley Values

- ▶ Beware: The Shapley value is NOT the difference in prediction when we would remove the feature from the model.
- ▶ Ex: Bike Rental



Shapley Values

- ▶ Advantages:
 - ▶ Difference between the prediction and the average prediction is **fairly** distributed among the feature values (could be more relevant from a legal point of view vs LIME for example)
 - ▶ Allows contrastive explanations
- ▶ Disadvantages:
 - ▶ Requires a lot of computing time
 - ▶ Can be misinterpreted
 - ▶ Shapley value: Given the current set of feature values, the contribution of a feature value to the difference between the actual prediction and the mean prediction
 - ▶ Not sparse, always use all the features
 - ▶ No prediction model like LIME
 - ▶ Need access to the data

SHAP (SHapley Additive exPlanations)

- ▶ Similar to the Shapley Value
- ▶ Difference: Shapley value explanation is represented as an additive feature attribution method, a linear model.
- ▶ Connects LIME and Shapley values

$$g(z') = \phi_0 + \sum_{j=1}^M \phi_j z'_j$$

- ▶ g is the explanation model, $z' \in \{0,1\}^M$ is the coalition vector, M is the maximum coalition size and ϕ_j is the feature attribution for a feature j (Shapley value)

SHAP (SHapley Additive exPlanations)

- ▶ KernelSHAP
- ▶ To find the SHAP values, we train a linear model:

$$L(\hat{f}, g, \pi_x) = \sum_{z' \in Z} [\hat{f}(h_x(z')) - g(z')]^2 \pi_x(z')$$

where

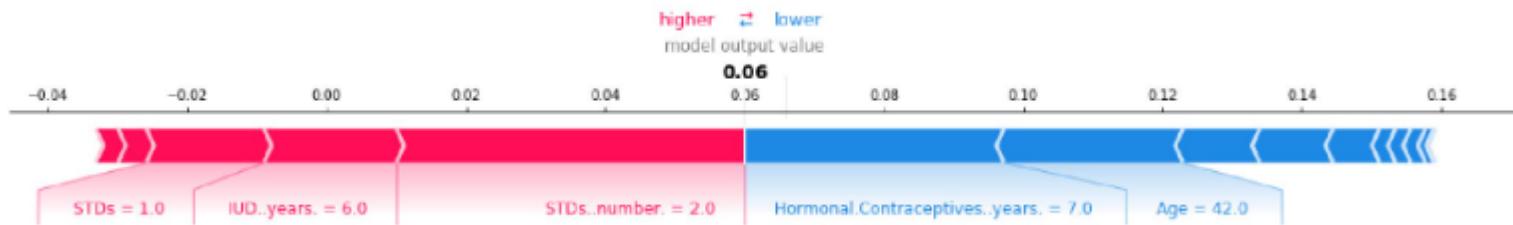
$$\pi_x(z') = \frac{(M-1)}{\binom{M}{|z'|} |z'| (M-|z'|)}$$

M is the maximum coalition size and |z'| is the number of present features in instance z'

- ▶ Small coalitions (few 1's) and large coalitions (many 1's) get the largest weights.

SHAP (SHapley Additive exPlanations)

- ▶ Ex: Cervical Cancer Dataset (RF trained model)
 - ▶ The average predicted probability is 0.066.
 - ▶ This woman has a low predicted risk of 0.06. Risk increasing effects such as STDs are offset by decreasing effects such as age.

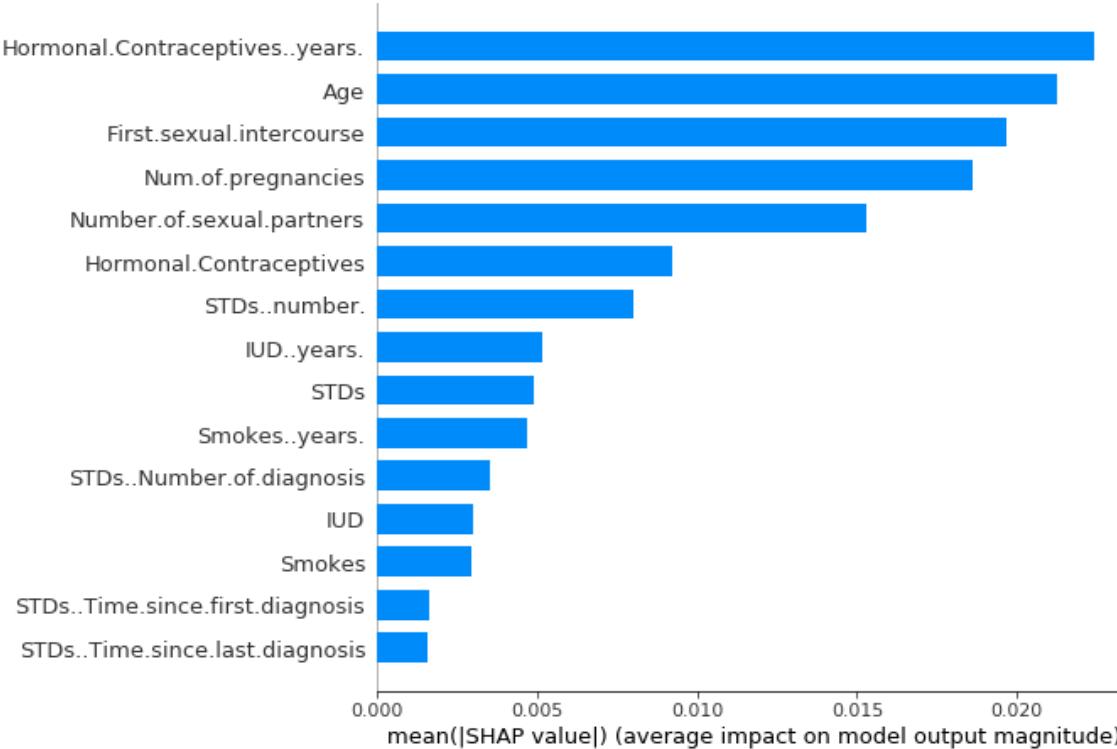


- ▶ This woman has a high predicted risk of 0.71. Age of 51 and 34 years of smoking increase her predicted cancer risk.



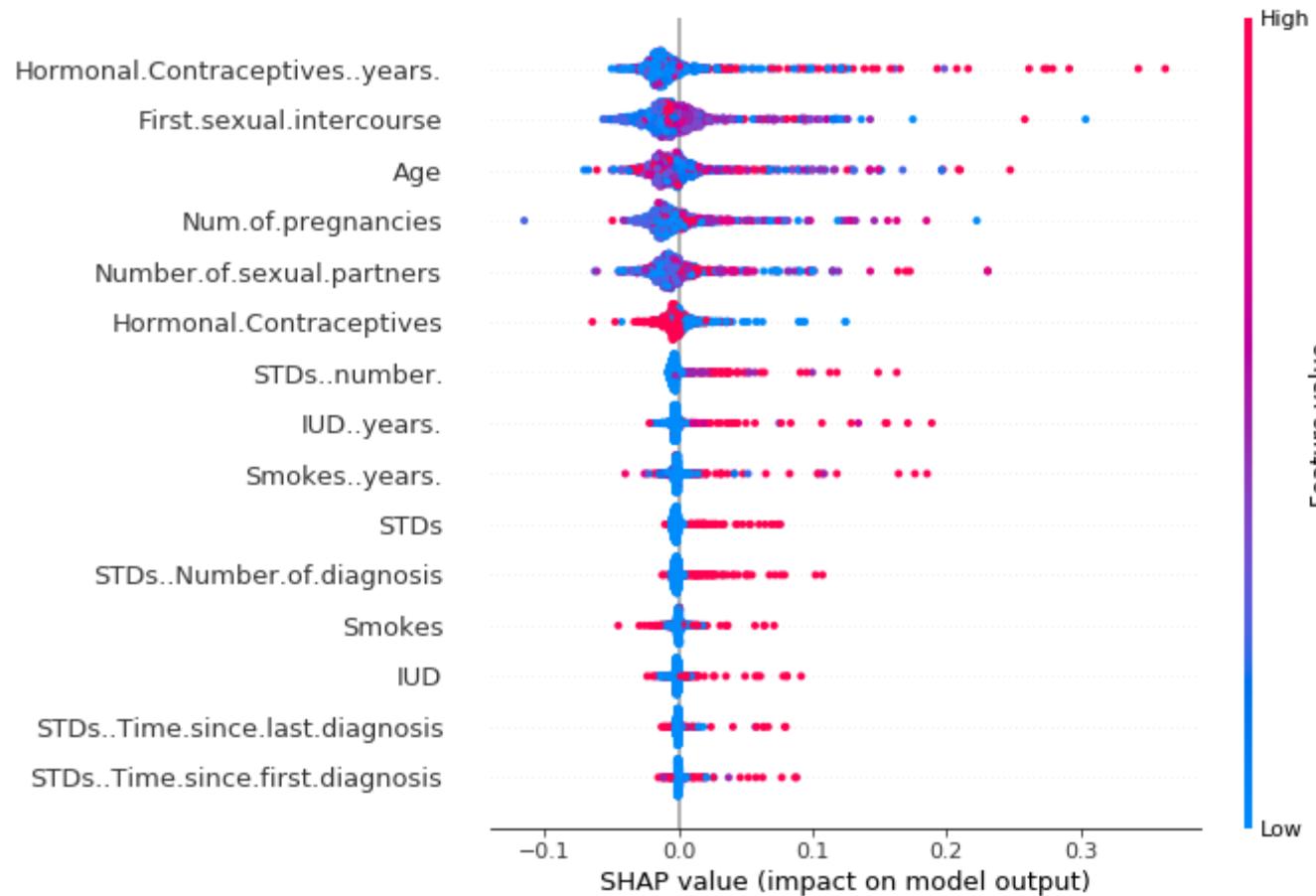
SHAP (SHapley Additive exPlanations)

- ▶ Shapley values can be combined into global explanations.
- ▶ SHAP Feature Importance
 - ▶ Features with large absolute Shapley values are important.
 - ▶ We average the absolute Shapley values per feature across the data



SHAP (SHapley Additive exPlanations)

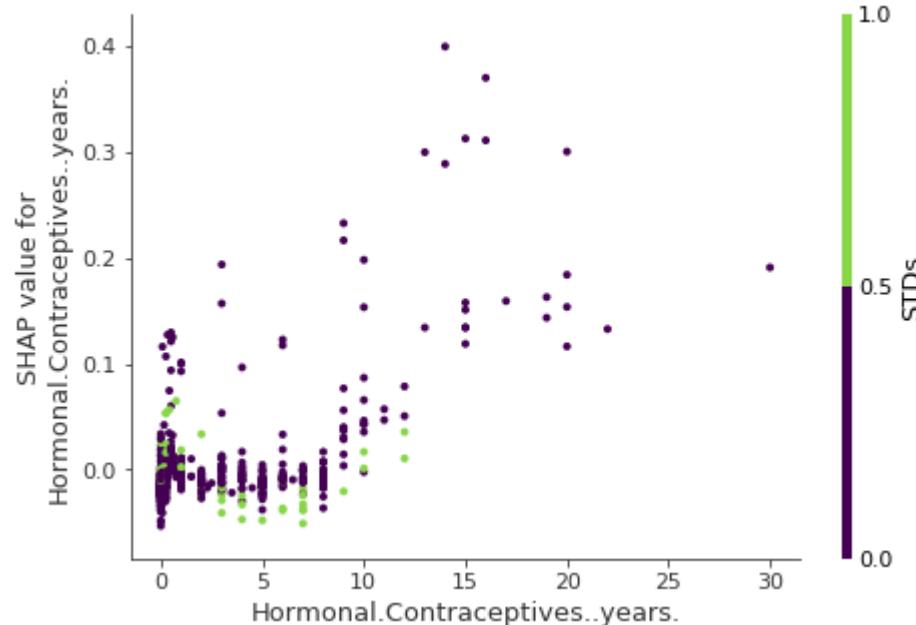
- ▶ SHAP Summary Plot
- ▶ Combines feature importance with feature effects



SHAP (SHapley Additive exPlanations)

▶ SHAP Interaction Values

- ▶ Additional combined feature effect after accounting for the individual feature effects
- ▶ In cases close to 0 years, the occurrence of a STD increases the predicted cancer risk. For more years on contraceptives, the occurrence of a STD reduces the predicted risk.



SHAP (SHapley Additive exPlanations)

- ▶ Advantages:
 - ▶ Prediction is **fairly** distributed among the feature values
 - ▶ Connects LIME and Shapley values
 - ▶ Could be used for global model interpretations
- ▶ Disadvantages:
 - ▶ KernelSHAP is slow: Impractical when you want to compute Shapley values for many instances
 - ▶ KernelSHAP ignores feature dependence.
 - ▶ Shapley values can be misinterpreted
 - ▶ Possible to create intentionally misleading interpretations with SHAP, which can hide biases

ÖZYEĞİN ÜNİVERSİTESİ

DS 530

Fairness and Interpretability

ENİS KAYIŞ

Living Together with Algorithms

- ▶ Decision making quality: Arbitrary, inconsistent, or faulty decision-making raises serious concerns for our quality of life.
- ▶ Data-driven decisions are more accurate than those based on intuition or expertise.
 - ▶ Ex: Automated underwriting of loans was found to be both more accurate and less racially disparate.
- ▶ Algorithms (ML) are increasingly becoming part of our everyday lives.

Machine Learning Process

- ▶ Two alternatives for automation:
 - ▶ Teaching a computer through explicit instruction in favor of a process
 - ▶ Learning by example: Exposing to many examples of images containing pre-identified objects
- ▶ Risks in learning from examples
 - ▶ Generalizing from examples
 - ▶ Process of induction: drawing general rules from specific examples
 - ▶ The hope is that we'll figure out how future cases are likely to be similar to past cases, even if they are not exactly the same.

Risks in ML

- ▶ What the algorithm needs?
 - ▶ Good examples:
 - ▶ a sufficiently large number of examples to uncover subtle patterns;
 - ▶ a sufficiently diverse set of examples to showcase the many different types of appearances that objects might take;
- ▶ ML models are only as reliable as the dataset on which it is trained.
- ▶ Being data-driven by no means ensures that it will lead to accurate, reliable, or fair decisions.

Risks in ML

- ▶ Historical examples of the relevant outcomes will reflect historical prejudices against certain social groups
 - ▶ Patterns in these data will replicate these very same dynamics.
- ▶ Human decision makers rarely try to maximize predictive accuracy at all costs
 - ▶ Also consider factors such as morality.
 - ▶ For example, although younger defendants are statistically more likely to re-offend, judges are loath to take this into account in deciding sentence lengths.
- ▶ Humans (unlike models) are also unlikely to make decisions that are obviously absurd
- ▶ Decision making systems that rely on ML might be unjust

Demographic disparities

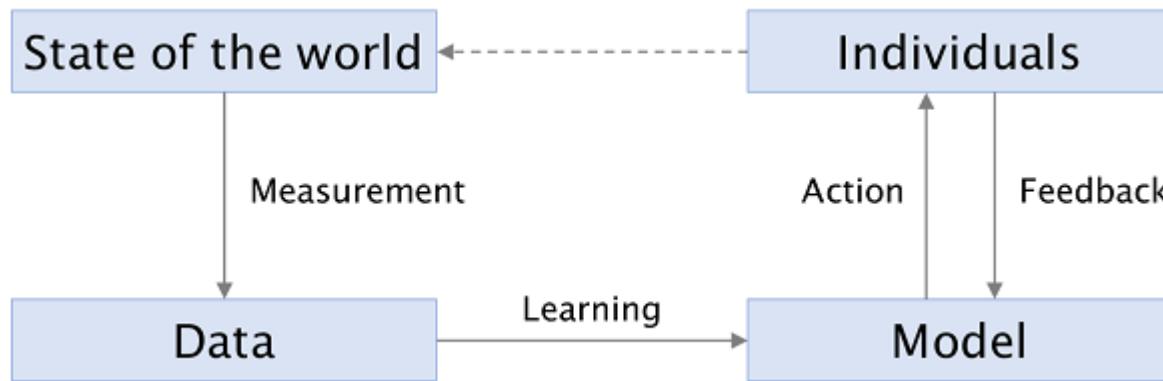
- ▶ Amazon uses a data-driven system to determine the neighborhoods in which to offer free same-day delivery.
 - ▶ Stark disparities in the demographic makeup of these neighborhoods: white residents were more than twice as likely as black residents to live in one of the qualifying neighborhoods.
- ▶ How to measure these inequalities precisely?
- ▶ What is the source of such disparities?
 - ▶ Intend?
- ▶ Two questions:
 - ▶ Are the disparities justified?
 - ▶ Are the disparities harmful?

Bias

- ▶ How to define bias?
 - ▶ Demographic disparities in algorithmic systems that are objectionable for societal reasons
 - ▶ Different understandings across communities
- ▶ Statistical Bias:
 - ▶ A statistical estimator is biased if its average value differs from the true value that it aims to estimate.
 - ▶ Well established theory behind the concept
- ▶ Other properties of a predictor: precision, recall, etc.
 - ▶ Well understood, with rich theory
- ▶ A new direction towards fairness
 - ▶ Is our goal to faithfully reflect the data?
 - ▶ Do we have an obligation to question the data, and to design our systems to conform to some notion of equitable behavior?

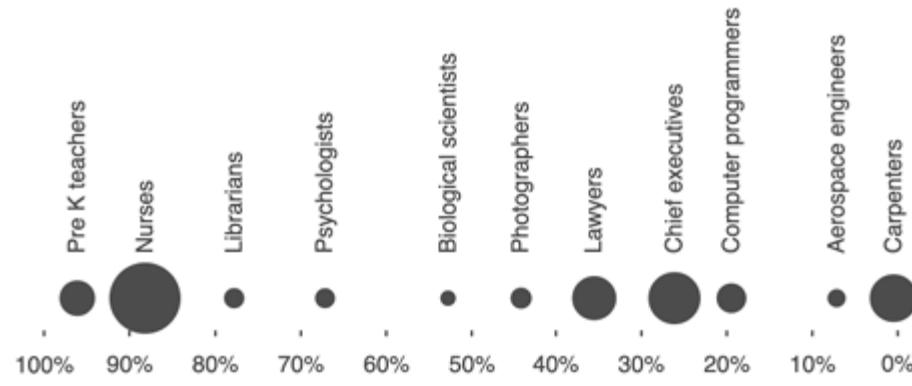
Machine Learning Loop

- ▶ How do disparities propagate through ML pipeline?
 - ▶ Measurement: A process by which the state of the world is reduced to a dataset
 - ▶ Learning: Turning that dataset into a model (weights, parameters) which summarizes patterns and makes generalizations
 - ▶ Action: Decisions we take based on predictions of the model
 - ▶ Feedback loop: exist in some systems (such as search engines)



State of The Society

- ▶ We are mainly interested in applications with datasets involving people.
 - ▶ Ex: Gender imbalance in occupations
 - ▶ While building a ML system that screens job candidates, we should be aware that this is the baseline.
- ▶ Why do these disparities exist?
 - ▶ Explicit or implicit reasons



A sample of occupations in decreasing order of the % of women. The area of the bubble is the number of workers.

Measurement Problems

- ▶ Measurement is fraught with subjective decisions and technical difficulties.
- ▶ Bias in the target variable is even more critical
 - ▶ Will the candidate be a good employee if hired? How to define a good employee? By sales volume? By performance reviews?
 - ▶ Creditworthiness: Is it a physical property or something we construct?
 - ▶ Physical Attractiveness

Measurement Problems

- ▶ Not all bad news: ML can help mitigate measurement bias
- ▶ Diagnoses are sometimes personalized by race.
 - ▶ Race is used as a crude proxy for ancestry and genetics, and sometimes environmental and behavioral factors
- ▶ We (data scientists) sometimes don't think about these steps
 - ▶ Someone else has already done those things.
- ▶ It's almost always too messy for algorithms to handle, hence the dreaded "data cleaning" step.
- ▶ But the messiness of the real world is a manifestation of a diverse world in which people don't fit neatly into categories.
- ▶ Being inattentive to these nuances can particularly hurt marginalized populations.

From data to models

- ▶ When we learn a model from problematic data, are disparities preserved, mitigated, or exacerbated?
- ▶ Predictive models trained with supervised learning methods are good at calibration:
 - ▶ Means that, we expect our models to faithfully reflect disparities found in the training data.
- ▶ Some patterns in the training data represent knowledge that we wish to mine using machine learning:
 - ▶ smoking is associated with cancer
- ▶ Other patterns represent stereotypes that we might wish to avoid learning:
 - ▶ girls like pink and boys like blue
- ▶ Algorithms have no general way to distinguish between these two types of patterns

From data to models

- When we build a statistical model of language from biased text, we expect the gender associations of occupation words to mirror real-world labor statistics.

The image displays two side-by-side screenshots of a web-based translation tool. Both screenshots show a top navigation bar with language selection buttons: English, Turkish, Spanish, Detect language, and a dropdown menu. A blue 'Translate' button is located on the right of each bar.

Top Screenshot (English to Turkish):

- Input: She is a doctor.
He is a nurse.
- Output: O bir doktor.
O bir hemşire.

Bottom Screenshot (Turkish to English):

- Input: O bir doktor.
O bir hemşire
- Output: He is a doctor.
She is a nurse ✓

Both screenshots include standard UI elements like a microphone icon for voice input, a text area with a character count (31/5000 and 28/5000), and a 'Translate' button.

From data to models

Turkish ▾ ↔ English ▾

O bir doktor × She is a doctor (feminine)

1 He is a doctor (masculine)

Open in Google Translate • Feedback

From data to models

- ▶ What if we simply withhold gender from the data?
- ▶ Not that simple, because of the problem of proxies
 - ▶ Other attributes in the data that might correlate with gender
 - ▶ Ex: Consider a programming job.
 - ▶ The age at which someone starts programming is correlated with gender.
 - ▶ Yet, we can't just get rid of proxies:
 - ▶ How long someone has been programming is a factor that gives us valuable information about their suitability
 - ▶ However, it also reflects the reality of gender stereotyping.

From data to models

- ▶ Another challenge: Sample size disparity
- ▶ If the trained dataset is sampled uniformly, then will have fewer data points about minorities.
- ▶ But ML works better when there's more data
- ▶ Thus, it will work less well for members of minority groups.
- ▶ When we develop machine-learning models, we typically only test their overall accuracy
 - ▶ a 5% error might hide the fact that a model performs terribly for a minority group.
- ▶ Reporting accuracy rates by group will help alert us to problems.
- ▶ If we're not careful, learning algorithms will generalize based on the majority culture, leading to a high error rate for minority groups.

The pitfalls of action

- ▶ Predictions made using a model that faithfully captures the patterns in the underlying data will inevitably have disparate error rates for different groups.
 - ▶ Understanding the properties of a prediction may not be sufficient, we need to understand population differences between the groups on which the predictions are applied (global vs local interpretability)
- ▶ An ethical decision making require, among other things, the ability to explain a prediction or decision.
- ▶ Models only reveals correlations, but we often use its predictions as if they reveal causation.
- ▶ Sometime the prediction could affect the outcome, and thus invalidates itself.
 - ▶ Ex: traffic congestion (if sufficiently many people choose their routes based on the prediction, then the route predicted to be clear will in fact be congested)
 - ▶ The effect can also work in the opposite direction:
 - ▶ The prediction might reinforce the outcome, resulting in feedback loops.

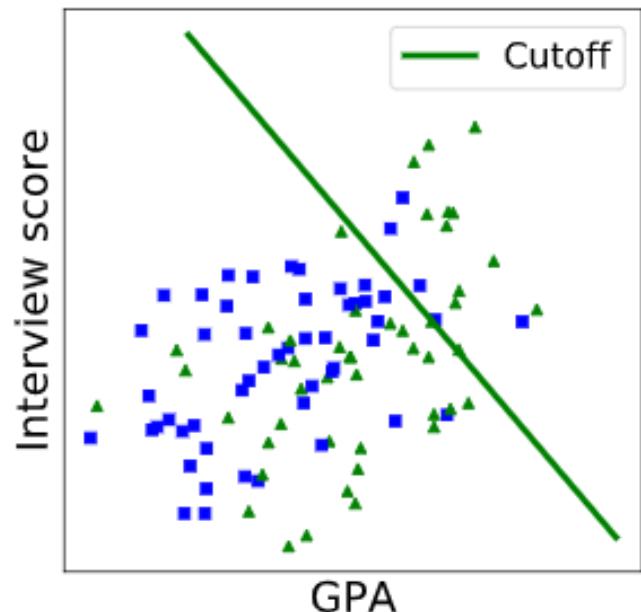
Feedback and feedback loops

- ▶ Many systems receive feedback when they make predictions
 - ▶ User clicks to search engine results, recommendation engine thumbs up/down
- ▶ Issues with feedback loops:
 - ▶ Self-fulfilling predictions: More police might be deployed to low-risk areas. Police in areas predicted to be high risk might be lowering their threshold for searching.
 - ▶ Predictions that affect the training set: Predictive policing will lead to arrests, records of which might be added to the algorithm's training set.
 - ▶ Predictions that affect the phenomenon and society at large: Prejudicial policing on a large scale, algorithmic or not, will affect society over time, contributing to the cycle of poverty and crime.

A Toy Example

- ▶ Goal: Hire high quality candidates
 - ▶ How to measure quality?
 - ▶ Average performance review score after two years at the firm
 - ▶ Input: college GPA and interview score
 - ▶ Let's use linear regression to predict the average job performance rating, and then use a cutoff based on the number of candidates we want to hire.

Two demographic groups, represented by triangles and squares



A Toy Example

- ▶ Any concerns?
 - ▶ How to enforce rigid categories of people?
 - ▶ The classifier did not take the group info (triangles and squares). Is it fair?
 - ▶ Our model gives same scores to otherwise-identical members of different groups, hence is not discriminatory at face value.
- ▶ But:
 - ▶ Are candidates from the two groups equally likely to be positively classified?
 - ▶ Of course not (the triangles are more likely to be selected than the squares)
 - ▶ Maybe managers who score the employees' performance might discriminate against one group?
 - ▶ Maybe the overall workplace might be less welcoming one group?
 - ▶ Maybe disparities before hiring (educational opportunities across groups)

A Toy Example

- ▶ How can we correct the demographic disparity?
 - ▶ Observe that GPA is correlated with the demographic attribute
 - ▶ Could we could simply omit that variable as a predictor?
- ▶ Pick different cutoffs so that candidates from both groups have the same probability of being hired (positive discrimination)
- ▶ Any issues with positive discrimination?
 - ▶ The pick-different-thresholds approach seems unsatisfying, as it uses the group attribute as the sole criterion for redistribution.
 - ▶ It does not account for the underlying reasons
- ▶ There is generally no mathematically principled way to know which cutoffs to pick
 - ▶ U.S. Equal Employment Opportunity Commission state that if the probability of selection for two groups differs by more than 20%, it might constitute a sufficient disparate impact to initiate a lawsuit
- ▶ Select different team members as diverse teams perform the best

ÖZYEĞİN ÜNİVERSİTESİ

DS 530

Fairness and Interpretability

ENİS KAYIŞ

Fairness in Classification

- ▶ Classification in two steps:
 1. Represent a population as a probability distribution.
 2. Apply statistics, specifically statistical decision theory, to the probability distribution that represents the population.
- ▶ Goal: determine a plausible value for an unknown target Y given observed covariates X .
 - ▶ the covariates X and target Y are jointly distributed random variables
- ▶ We observe the covariates X and make a guess $\hat{Y} = f(X)$ based on what we observed
 - ▶ $f(\cdot)$: classifier
 - ▶ We are not interested in the classifier but the random variable \hat{Y}
 - ▶ Assume: No feedback loop!

Classification

- ▶ What are the properties of a good classifier?
 - ▶ Classification Accuracy: $P(Y = \hat{Y})$
 - ▶ Classification Error: $P(Y \neq \hat{Y})$
 - ▶ Ex: A classifier that always predicts no traffic fatality in the next year might have high accuracy on any given individual
 - ▶ High accuracy, but of little value
 - ▶ There are others:

Common classification criteria		
Event	Condition	Resulting notion ($\mathbb{P}\{\text{event} \mid \text{condition}\}$)
$\hat{Y} = 1$	$Y = 1$	True positive rate, recall
$\hat{Y} = 0$	$Y = 1$	False negative rate
$\hat{Y} = 1$	$Y = 0$	False positive rate
$\hat{Y} = 0$	$Y = 0$	True negative rate

Classification

▶ Alternative via swapping event and condition

Additional classification criteria

Event	Condition	Resulting notion ($\mathbb{P}\{\text{event} \mid \text{condition}\}$)
$Y = 1$	$\hat{Y} = 1$	Positive predictive value, precision
$Y = 0$	$\hat{Y} = 0$	Negative predictive value

▶ An optimal classifier is any classifier that minimizes the expected loss

$$\mathbb{E}[\ell(\hat{Y}, Y)]$$

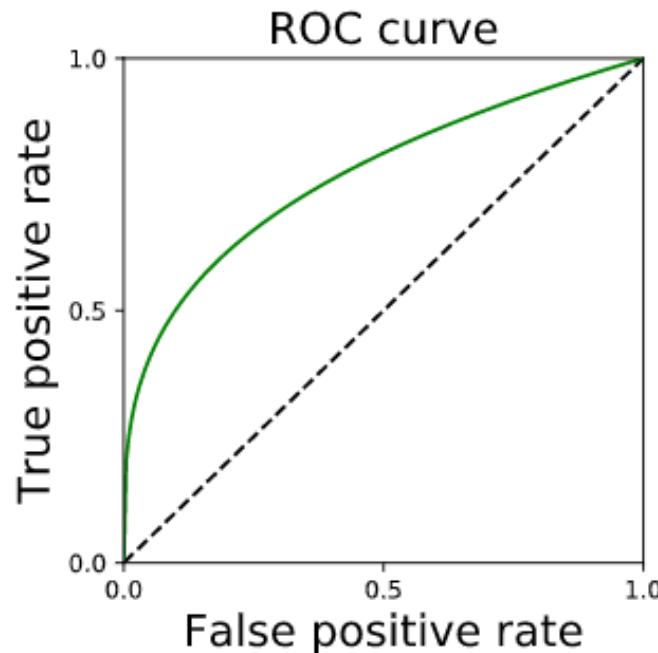
- ▶ As an example, choose the losses $I(0,1)=I(1,0)=1$ and $I(1,1)=I(0,0)=0$.
- ▶ Given this loss function, one can show that

Fact. *The optimal predictor minimizing classification error satisfies*

$$\hat{Y} = f(X), \quad \text{where} \quad f(x) = \begin{cases} 1 & \text{if } \mathbb{P}\{Y = 1 \mid X = x\} > 1/2 \\ 0 & \text{otherwise.} \end{cases}$$

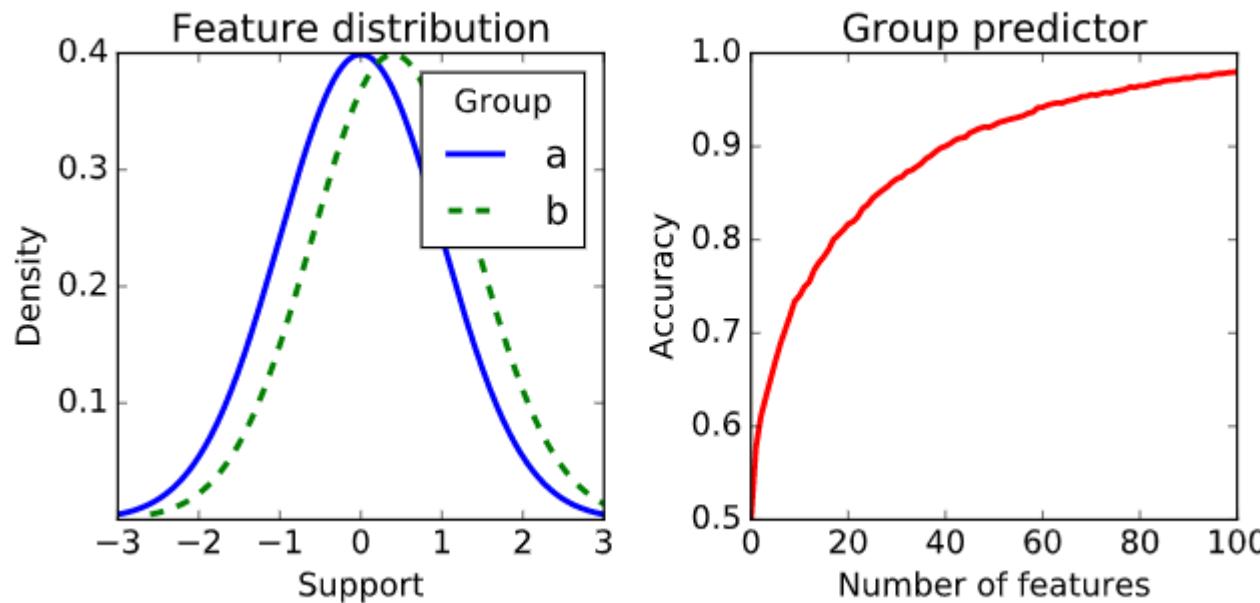
Classification

- ▶ ROC curve:
 - ▶ By varying the classification threshold from 0 to 1, we can find a curve where the axes correspond to true positive rate and false positive rate.
 - ▶ An area of 0.5 corresponds to random guessing, and an area of 1 corresponds to perfect classification.



No fairness through unawareness

- ▶ Idea: Remove sensitive attributes to have fairness
 - ▶ Unfortunately, this can be ineffective and even harmful.



- ▶ In large feature spaces sensitive attributes are generally redundant given the other features.

Statistical non-discrimination criteria

- ▶ Let's define the absence of discrimination in terms of statistical expressions
- ▶ Statistical non-discrimination criteria depends on:
 - ▶ sensitive attribute A
 - ▶ target variable Y ,
 - ▶ the classifier \hat{Y} ,
 - ▶ and features X .
- ▶ Now, we can unambiguously decide whether or not a criterion is satisfied by looking at the joint distribution of these random variables.

Statistical non-discrimination criteria

- ▶ Idea: equalize some group-dependent statistical quantity across groups defined by the different settings of A.
- ▶ Ex: Equalize acceptance rates across all groups

$$\mathbb{P}\{\hat{Y} = 1 \mid A = a\} = \mathbb{P}\{\hat{Y} = 1 \mid A = b\}$$

- ▶ Equalize three fundamental statistics
 - Acceptance rate $\mathbb{P}\{\hat{Y} = 1\}$ of a classifier \hat{Y}
 - Error rates $\mathbb{P}\{\hat{Y} = 0 \mid Y = 1\}$ and $\mathbb{P}\{\hat{Y} = 1 \mid Y = 0\}$ of a classifier \hat{Y}
 - Outcome frequency given score value $\mathbb{P}\{Y = 1 \mid R = r\}$ of a score R

Demographic parity

- ▶ Also known as statistical parity

$$\mathbb{P}\{\hat{Y} = 1 \mid A = a\} = \mathbb{P}\{\hat{Y} = 1 \mid A = b\}$$

- ▶ requires the acceptance rate ($Y=1$) to be the same in all groups
- ▶ Could be relaxed somewhat

$$\mathbb{P}\{\hat{Y} = 1 \mid A = a\} \geq \mathbb{P}\{\hat{Y} = 1 \mid A = b\} - \epsilon$$

- ▶ Alternatively

$$\frac{\mathbb{P}\{\hat{Y} = 1 \mid A = a\}}{\mathbb{P}\{\hat{Y} = 1 \mid A = b\}} \geq 1 - \epsilon$$

Statistical non-discrimination criteria

- ▶ Demographic parity has nice mathematical properties.
- ▶ Limitations:
 - ▶ Selective hiring across groupings to equalize acceptance rates
 - ▶ Insufficiency of training data as it relates to minority groups

Statistical non-discrimination criteria

- ▶ Equalized Odds (Error rate parity)
 - ▶ A predictor \hat{Y} satisfies **equalized odds** with respect to protected attribute A and outcome Y , if \hat{Y} and A are independent conditional on Y .
$$\mathbb{P}\{\hat{Y} = 1 \mid Y = 1, A = a\} = \mathbb{P}\{\hat{Y} = 1 \mid Y = 1, A = b\}$$
$$\mathbb{P}\{\hat{Y} = 1 \mid Y = 0, A = a\} = \mathbb{P}\{\hat{Y} = 1 \mid Y = 0, A = b\}$$
 - ▶ The probability of a person in the positive class being correctly assigned a positive outcome and the probability of a person in a negative class being incorrectly assigned a positive outcome should both be the same for the protected and unprotected group members.
 - ▶ Measuring and reporting group specific error rates can create an incentive to collect better datasets and build better models.

Statistical non-discrimination criteria

- ▶ Predictive Parity:
 - ▶ A predictor Y satisfies **predictive parity** if

$$\mathbb{P}\{Y = 1 \mid R = r, A = a\} = \mathbb{P}\{Y = 1 \mid R = r, A = b\}$$

Example: Different Metrics

Predictive Parity: $P(Y=1|A=0, \hat{Y}=y) = P(Y=1|A=1, \hat{Y}=y)$, $\hat{y} \in \{0,1\}$

Equalized Odd: $P(\hat{Y}=1|A=0, Y=y) = P(\hat{Y}=1|A=1, Y=y)$, $\hat{y} \in \{0,1\}$

Example: confusion matrix for credit risk

WOMEN (w)

Predicted score (\hat{Y})	Actual (Y)	
	Default (1)	Not (0) Default
High + (1)	60	20
Low - (0)	6	14

MEN (m)

Predicted score (\hat{Y})	Actual (Y)	
	Default (1)	Not (0) Default
High +	16	5
Low -	22	57

Predictive Parity:

$$P(Y=1|A=w, \hat{Y}=1) = \frac{60}{60+20} = 0.75$$

=PPV-Pos predictive value

$$P(Y=1|A=m, \hat{Y}=1) = \frac{16}{16+5} = 0.76$$

Equalized Odd:

$$P(\hat{Y}=1|A=w, Y=0) = \frac{20}{20+14} = 0.59$$

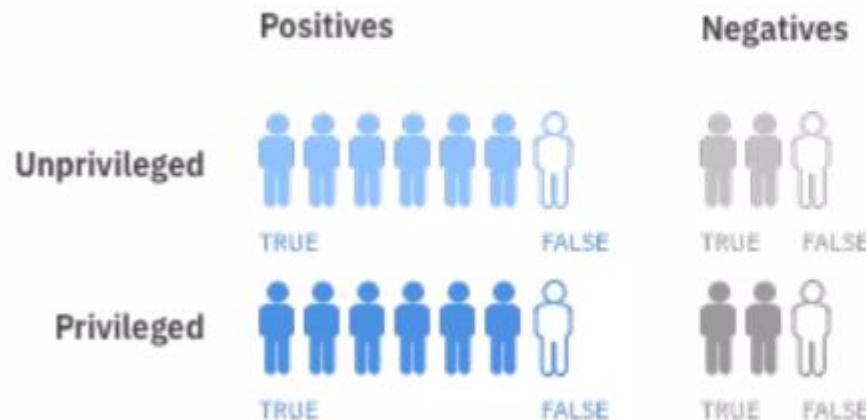
=FPR-false pos rate

$$P(\hat{Y}=1|A=m, Y=0) = \frac{5}{5+57} = 0.08$$

Fairness

► Group Fairness metrics

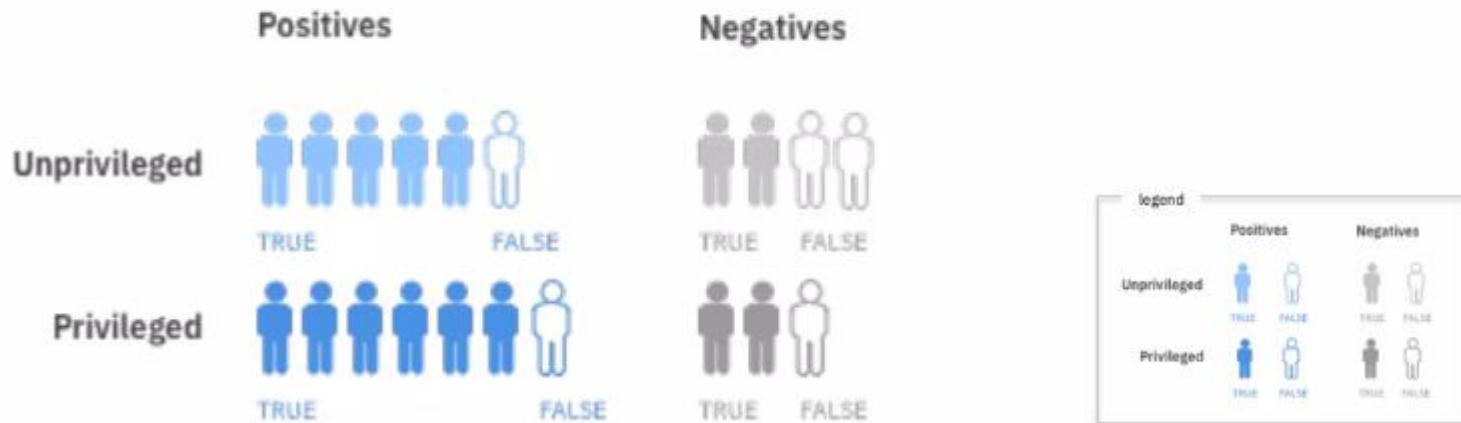
situation 1



Fairness

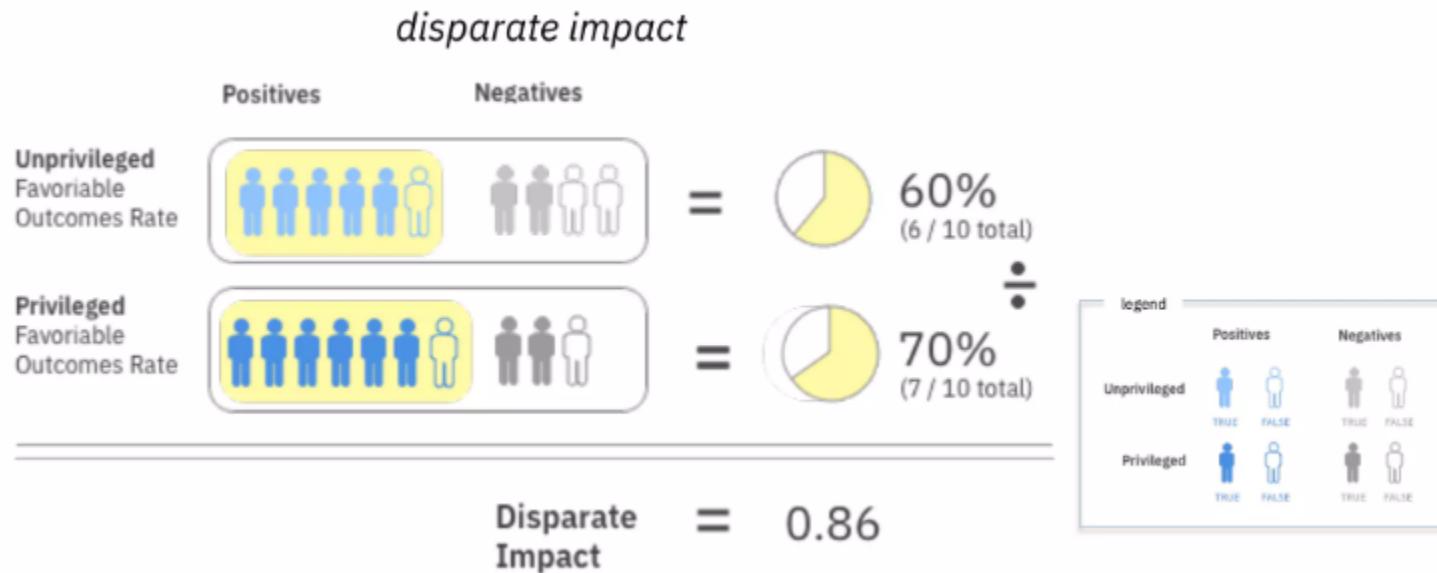
► Group Fairness metrics

situation 2



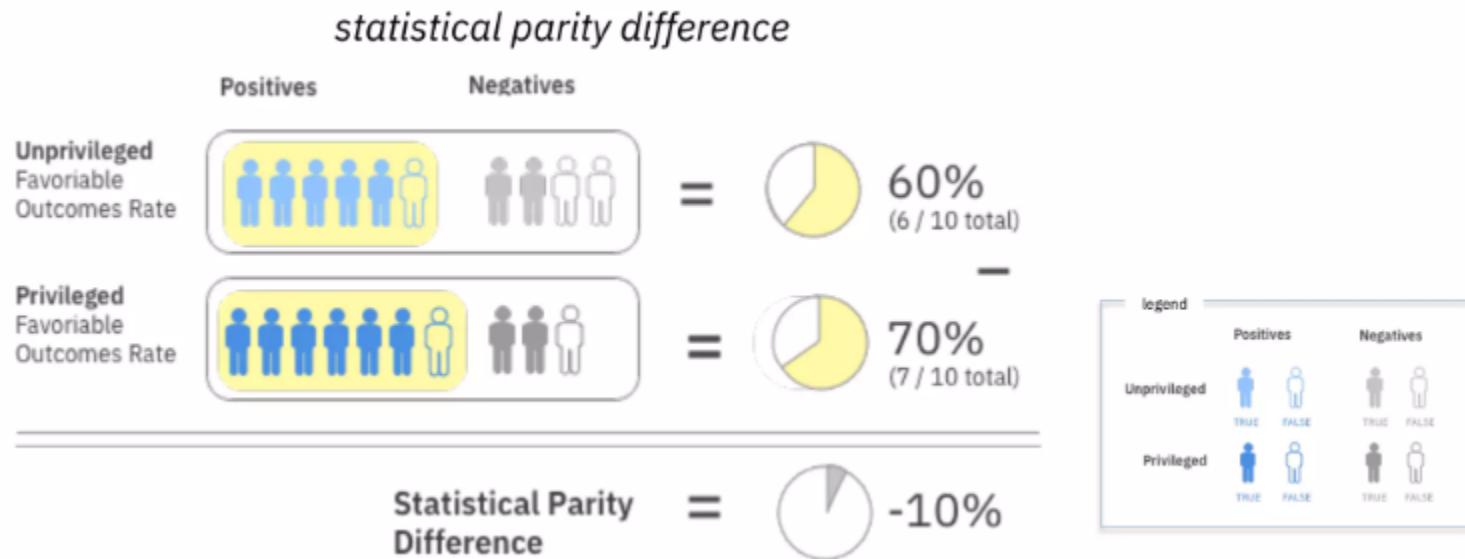
Fairness

► Group Fairness metrics



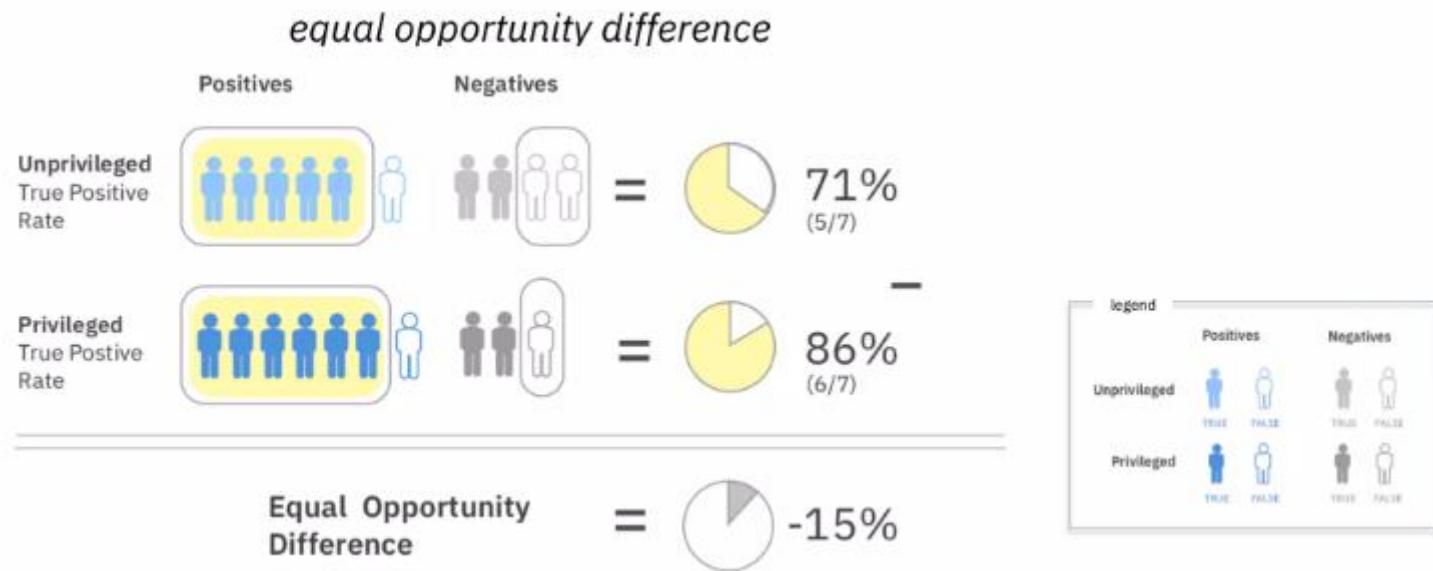
Fairness

► Group Fairness metrics



Fairness

► Group Fairness metrics



How to satisfy a non-discrimination criterion

- ▶ **Pre-processing:** Adjust the feature space to be uncorrelated with the sensitive attribute.
- ▶ **In-processing:** Work the constraint into the optimization process that constructs a classifier from training data.
- ▶ **Post-processing:** Adjust a learned classifier so as to be uncorrelated with the sensitive attribute.

Pre-processing

- ▶ Idea: Transform a feature space into a representation that as a whole is independent of the sensitive attribute
- ▶ Agnostic to the training models used down the pipeline

In-processing

- ▶ Training with fairness as a constraint during training
- ▶ We need data and the training model
- ▶ Specific to the training models used, hence is not model agnostic

Post-processing

- ▶ Idea: Take a trained classifier and adjust it possibly depending on the sensitive attribute and additional randomness in such a way that fairness is achieved.
- ▶ Works for any black-box classifier regardless of its inner workings.
- ▶ No need for re-training, which is useful in cases where the training pipeline is complex and time-consuming.
- ▶ The only available option when we have access only to a trained model with no control over the training process.
- ▶ Cons: The resulting classifier uses group membership quite explicitly by setting different acceptance thresholds for different groups (positive discrimination).

Case study: Credit scoring

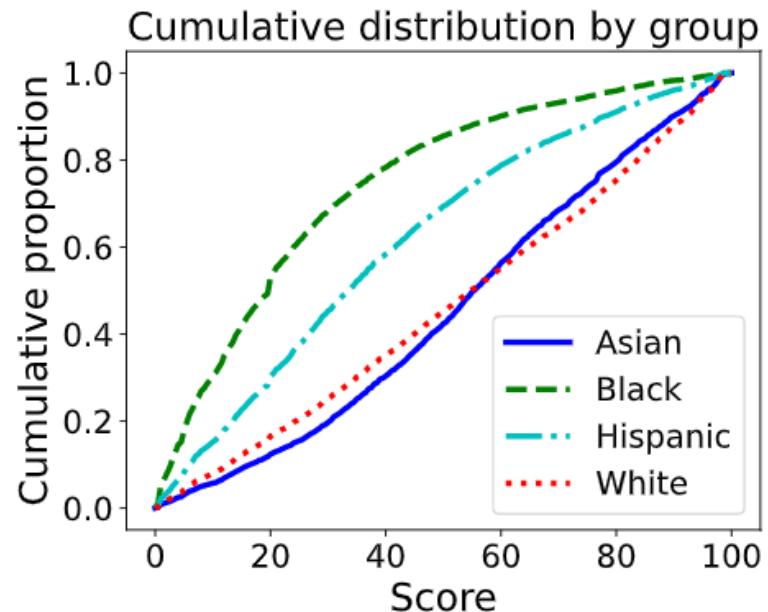
- ▶ Credit scores are widely used in the United States to allocate credit
- ▶ Support lending decisions by giving an estimate of the risk that a loan applicant will default
- ▶ In the US, three major credit-reporting agencies collect data on various lendees.
- ▶ These agencies are for-profit organizations that each offer risk scores based on the data they collected.
- ▶ The Equal Credit Opportunity Act makes it unlawful for any creditor to discriminate against any applicant the basis of race, color, religion, national origin, sex, marital status, or age.

Case study: Credit scoring

- ▶ Credit scores are widely used in the United States
- ▶ The following are summary for TransUnion's TransRisk score.
 - ▶ The scores for the study are normalized to vary from 0 to 100, with 0 being least creditworthy
 - ▶ Race is self-reported

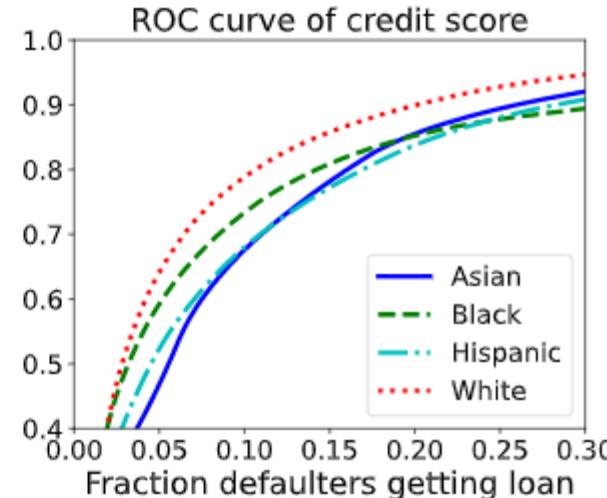
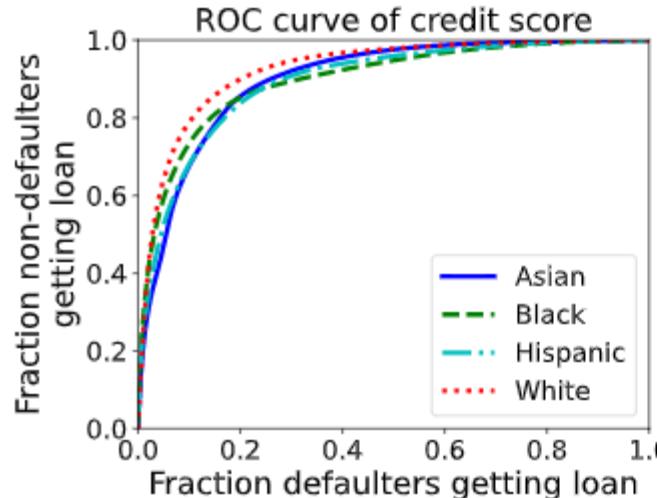
Credit score distribution by ethnicity

Race or ethnicity	Samples with both score and outcome
White	133,165
Black	18,274
Hispanic	14,702
Asian	7,906
Total	174,047



Case study: Credit scoring

- ▶ How to define the outcome (default)?
 - ▶ Defaults vary in the amount of debt recovered, and the amount of time given for recovery.
 - ▶ Define: (the) measure is based on the performance of new or existing accounts and measures whether payment have been late 90 days or more on one or more of the accounts or had a public record item or a new collection agency account during the performance period.

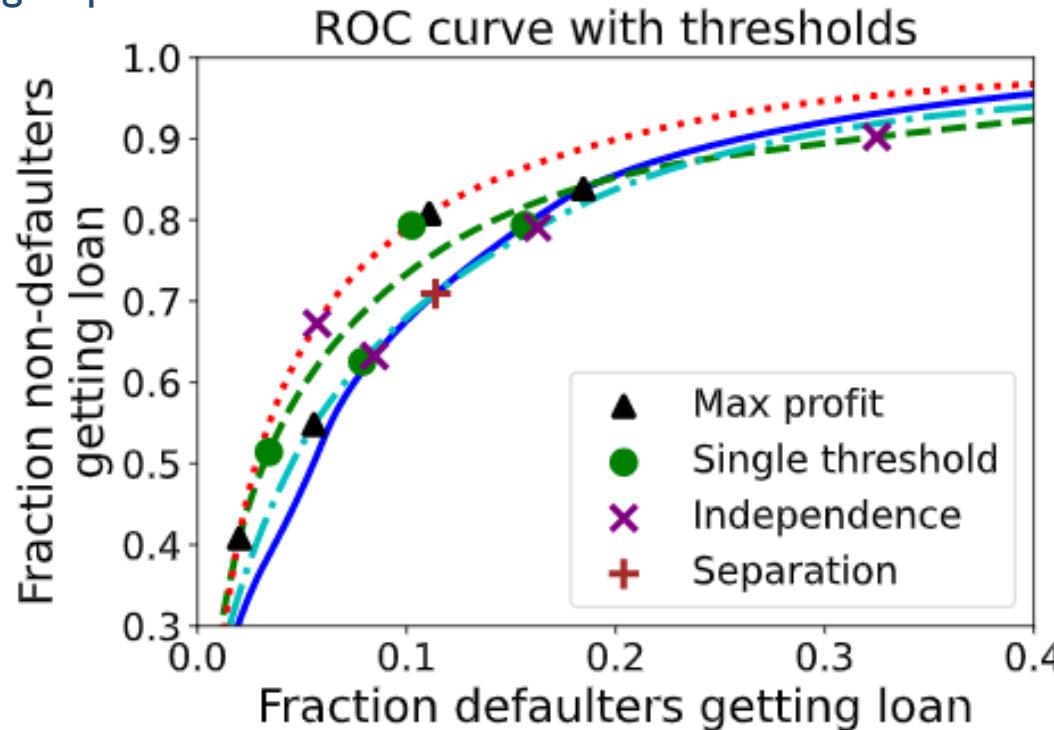


Case study: Credit scoring

- ▶ Consider four different classification strategies
 - ▶ **Maximum profit:** Pick possibly group-dependent score thresholds in a way that maximizes profit.
 - ▶ **Single threshold:** Pick a single uniform score threshold for all groups in a way that maximizes profit.
 - ▶ **Independence:** Achieve an equal acceptance rate in all groups. Subject to this constraint, maximize profit.
 - ▶ **Separation:** Achieve an equal true/false positive rate in all groups. Subject to this constraint, maximize profit.
- ▶ Profit: Assume that the cost of a false positive is 6 times greater than the return on a true positive
 - ▶ Defaults are much costlier than profit from loan interest

Case study: Credit scoring

- ▶ The true positive rate achieved by max profit for the Asian group is twice of what it is for the Black group.
- ▶ The separation criterion results in the same trade-off in all groups.
- ▶ Independence equalizes acceptance rate but leads to widely different tradeoffs.
 - ▶ Black group has a false positive rate more than three times higher than the false positive rate of the Asian group.



Fairness

- ▶ Group Fairness metrics
- ▶ Demo Application:AI Fairness 360 Web Application
 - ▶ <https://aif360.res.ibm.com/data>
 - ▶ Check out the demos using different datasets