Exploring counterfactual explanations for classification and regression trees

Suryabhan Singh Hada and Miguel Á. Carreira-Perpiñán
{shada, mcarreira-perpinan}@ucmerced.edu

Dept. Computer Science and Engineering
University of California, Merced

XKDD, September 2021

Counterfactual explanations

- A counterfactual explanation seeks the minimal change to a given feature vector that will change a classifier's decision in a prescribed way.
- Consider following example:
  - Loan application is denied by bank (classifier).
  - Applicant ask: "what should I change to get it approved"?
  - Bank replies: "If annual income had been \$45,000 instead of \$30,000, the loan would have been approved".
- Counterfactual explanation is important to interpret a black-box decision for a given instance.
- Mathematically, the problem can be formulated as: given a source instance $\overline{\mathbf{x}}$, target class y and a classifier T, find the closest instance $\mathbf{x}$ to $\overline{\mathbf{x}}$ such that $\mathbf{x}$ is classified y $(T(\mathbf{x}) = y)$.
- Here, we focus on decision tree classifiers (also regression trees).

Decision trees are important machine learning models

- Decision trees are important, particularly in applications where interpretability is desirable, such as business, law, and medicine.
- Decision trees and forest regularly appeared as the most used machine learning models in surveys from `Kaggle` and `kdnuggets`.
- Thus, solving counterfactual explanations for decision trees is important in practice.

Tree alternating optimization (TAO)

- Traditionally, decision trees have been trained with a recursive partition procedure, such as CART and C4.5. However, this produces sub-optimal trees and does not work well with oblique trees (having hyperplane split).
- Tree alternating optimization (TAO) is a recently proposed algorithm that can achieve highly accurate oblique or axis-aligned trees.
- TAO can also train sparse oblique trees (having hyperplane splits with few nonzero weights) that are not only highly accurate but also small (shallow and with few nodes) and very interpretable.
- Stronger predictive power of sparse oblique decision trees together with their interpretability and fast inference makes them useful for other uses, such as in understanding deep neural networks or compressing deep neural networks.

Learning a single tree with TAO: general formulation

- TAO finds good approximate optima of an objective function over a tree with predetermined structure and it applies to trees beyond axis-aligned splits.
- We consider trees whose nodes make hard decisions (not soft trees). Optimizing such trees is difficult because they are not differentiable. Assuming a tree structure $\mathbf{T}$ is given (say, binary complete of depth Δ), consider the following optimization problem over its parameters:

$$E(\boldsymbol{\Theta}) = \sum_{n=1}^{N} L(\mathbf{y}_n, \mathbf{T}(\mathbf{x}_n; \boldsymbol{\Theta})) + \lambda \sum_{\text{nodes } i} \phi_i(\boldsymbol{\theta}_i)$$

given a training set $\{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$. $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_i\}$ is a set of parameters of all tree nodes. The loss function $L(\mathbf{y}, \mathbf{z})$ is 0/1 classification loss (although it is possible to use other losses, such as logistic or hinge). The regularization term ϕ_i (e.g. ℓ_1 norm) penalizes the parameters $\boldsymbol{\theta}_i$ of each node.

Source: Carreira-Perpiñán & Zharmagambetov, FODS 2020

Learning a single tree with TAO: separability of nodes

TAO algorithm is based on 3 theorems: separability condition, reduced problem over a leaf, reduced problem over a decision node. Here, we briefly mention them.

1. Separability condition

Consider any pair of nodes i and j. Assume the parameters of all other nodes ($\boldsymbol{\Theta}_{\text{rest}}$) are fixed. If nodes i and j are not descendants of each other, then $E(\boldsymbol{\Theta})$ can be rewritten as:

$$E(\boldsymbol{\Theta}) = E_i(\boldsymbol{\theta}_i) + E_j(\boldsymbol{\theta}_j) + E_{\text{rest}}(\boldsymbol{\Theta}_{\text{rest}})$$

In other words, the separability condition states that any set of non-descendant nodes of a tree can be optimized independently. Note that $E_{\text{rest}}(\boldsymbol{\Theta}_{\text{rest}})$ can be treated as a constant since we fix $\boldsymbol{\Theta}_{\text{rest}}$.

Source: Carreira-Perpiñán & Zharmagambetov, FODS 2020

Learning a single tree with TAO: leaves

All leaves are non-descendants of each others. Therefore, we can optimize over each of them independently (according to separability condition).

2. Reduced problem over a leaf

Assume node i is a leaf, then the optimization of $E(\boldsymbol{\Theta})$ over $\boldsymbol{\theta}_i$ can be equivalently rewritten as:

$$\min_{\boldsymbol{\theta}_i} E_i(\boldsymbol{\theta}_i) = \sum_{n \in \mathcal{R}_i} L(\mathbf{y}_n, \mathbf{g}_i(\mathbf{x}_n; \boldsymbol{\theta}_i)) + \alpha\, \phi_i(\boldsymbol{\theta}_i)$$

The reduced set $\mathcal{R}_i$ contains the training instances that reach leaf i. Each leaf i has a predictor function $\mathbf{g}_i(\mathbf{x}; \boldsymbol{\theta}_i)$: $\mathbb{R}^D \rightarrow \mathbb{C}$ (we use a constant or linear classifier) that produces the output class. Therefore, solving the reduced problem over a leaf i amounts to fitting the leaf's predictor $\mathbf{g}_i$ to the instances in its reduced set to minimize the original loss (e.g. misclassification error).

Source: Carreira-Perpiñán & Zharmagambetov, FODS 2020

Learning a single tree with TAO: decision nodes

To optimize decision (internal) nodes, we again consider a set of non-descendant (e.g. all nodes at the same depth) nodes. Optimizing over the parameters of one decision node is given by the following theorem.

3. Reduced problem over a decision node

If i is a decision node, the optimization of $E(\boldsymbol{\Theta})$ over $\boldsymbol{\theta}_i$ can be equivalently rewritten as:

$$\min_{\boldsymbol{\theta}_i} E_i(\boldsymbol{\theta}_i) = \sum_{n \in \mathcal{R}_i} l_{in}(f_i(\mathbf{x}_n; \boldsymbol{\theta}_i)) + \alpha\, \phi_i(\boldsymbol{\theta}_i)$$

where $\mathcal{R}_i$ is the reduced set of node i and (assuming binary trees) $f_i(\mathbf{x}; \boldsymbol{\theta}_i)$: $\mathbb{R}^D \rightarrow \{\texttt{left}, \texttt{right}\}$ is a decision function at node i which sends instance $\mathbf{x}_n$ to the corresponding child of i. We consider oblique trees, having hyperplane decision functions "go to right if $\mathbf{w}_i^T \mathbf{x} + w_{i0} \geq 0$" (where $\boldsymbol{\theta}_i = \{\mathbf{w}_i, w_{i0}\}$). $l_{in}(\cdot)$ is the loss incurred if $\mathbf{x}_n$ chooses the right or left subtree.

Source: Carreira-Perpiñán & Zharmagambetov, FODS 2020

Learning a single tree with TAO: decision nodes (cont.)

The reduced problem over a decision node can be equivalently rewritten as a weighted 0/1 loss binary classification problem on the node's reduced set instances:

$$\min_{\boldsymbol{\theta}_i} E_i(\boldsymbol{\theta}_i) = \sum_{n \in \mathcal{R}_i} \overline{L}_{in}(\overline{y}_{in}, f_i(\mathbf{x}_n; \boldsymbol{\theta}_i)) + \alpha\, \phi_i(\boldsymbol{\theta}_i)$$

where the weighted 0/1 loss $\overline{L}_{in}(\overline{y}_{in}, \cdot)$ for instance $n \in \mathcal{R}_i$ is defined as $\overline{L}_{in}(\overline{y}_{in}, y) = l_{in}(y) - l_{in}(\overline{y}_{in})\ \forall y \in \{\texttt{left}, \texttt{right}\}$, where $\overline{y}_{in} = \arg\min_y l_{in}(y)$ is a "pseudolabel" indicating a child which gives the lowest value of the loss L for instance $\mathbf{x}_n$ under the current tree. For hyperplane nodes, this is NP-hard, but can be approximated by using a convex surrogate loss (we use the logistic loss). Hence, if ϕ_i is an ℓ_1 norm, this requires solving an ℓ_1-regularized logistic regression.

Source: Carreira-Perpiñán & Zharmagambetov, FODS 2020

Pseudocode for TAO

TAO repeatedly alternates optimizing over sets of nodes by training a (binary) classifier in the decision nodes and a (multiclass) classifier in the leaves, while monotonically decreasing the obj. function $E(\boldsymbol{\Theta})$.

```
**input** training set; initial tree $\mathbf{T}(\cdot;\boldsymbol{\Theta})$ of depth $\Delta$
$\mathcal{N}_0,\dots,\mathcal{N}_\Delta \leftarrow$ nodes at depth $0,\dots,\Delta$, respectively
$\mathcal{R}_1 \leftarrow \{1,\dots,N\}$
**repeat**
  **for** $d = 0$ **to** $\Delta$
    **parfor** $i \in \mathcal{N}_d$
      **if** $i$ is a leaf **then**
        $\boldsymbol{\theta}_i \leftarrow$ train classifier $\mathbf{g}_i$ on reduced set $\mathcal{R}_i$
      **else**
        $\boldsymbol{\theta}_i \leftarrow$ train decision function $f_i$ on $\mathcal{R}_i$
        compute the reduced sets of each child of $i$
**until** stop
prune dead subtrees of $\mathbf{T}$
**return** $\mathbf{T}$
```

Source: Carreira-Perpiñán & Zharmagambetov, FODS 2020

Basic formulation of the counterfactual explanation problem

Given an input instance $\overline{\mathbf{x}} \in \mathbb{R}^D$, classifier T, and target class y

$$\min_{\mathbf{x}\in\mathbb{R}^D} E(\mathbf{x};\overline{\mathbf{x}}) \quad \text{s.t.} \quad T(\mathbf{x}) = y,\ \mathbf{c}(\mathbf{x}) = \mathbf{0},\ \mathbf{d}(\mathbf{x}) \geq \mathbf{0}$$

where $E(\mathbf{x};\overline{\mathbf{x}})$ is a cost of changing features of $\overline{\mathbf{x}}$, and $\mathbf{c}(\mathbf{x})$ and $\mathbf{d}(\mathbf{x})$ are problem-dependent equality and inequality constraints.

How to solve this optimization problem:

- If T is differentiable with respect to $\mathbf{x}$ the problem can be solved using gradient based methods, for example if T is a neural net.
- With decision trees T is not differentiable, this makes problem nondifferentiable and gradient-based methods are not applicable. However, this problem can be solved exactly and efficiently.

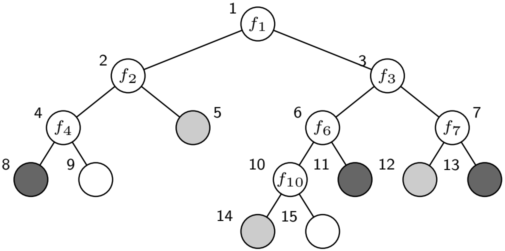
Counterfactual solution for decision tree

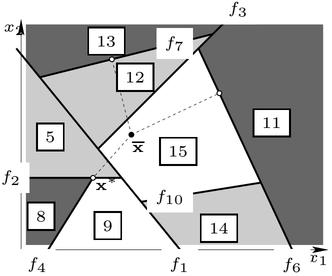
- A trained decision tree can be regarded as the partition of the input space into disjoint regions, where each region corresponds to one leaf.

- Therefore, finding the closest instance to the source instance having a desired target label can be done by finding closest instance in each leaf region, and picking the best among them.

- For each leaf, the region is defined by a polytope that acts as linear constraints. So, finding the counterfactual in a leaf becomes a quadratic/linear program that can be solved effectively.

Counterfactual explanations in decision trees





- The tree has L leaves, and it partitions the input space into L polytopes.
- Each polytope is defined by the intersection of arbitrary hyperplanes ("if $\mathbf{w}_i^T\mathbf{z} + b_i \geq 0$ then go to right child, else go to left child") found in the path from root to a leaf.
- Source instance $\overline{\mathbf{x}}$ is in white class.
- The counterfactual instance subject to being in dark grey class is $\mathbf{x}^*$, which is closest to $\overline{\mathbf{x}}$.

Counterfactual explanations in decision trees

Original problem:

$$\min_{\mathbf{x}\in\mathbb{R}^D} E(\mathbf{x};\overline{\mathbf{x}}) \quad \text{s.t.} \quad T(\mathbf{x}) = y,\ \mathbf{c}(\mathbf{x}) = \mathbf{0},\ \mathbf{d}(\mathbf{x}) \geq \mathbf{0}. \tag{1}$$

Theorem

Problem (1) *is equivalent to:*

$$\min_{i\in\mathcal{L}} \min_{\mathbf{x}\in\mathbb{R}^D} E(\mathbf{x};\overline{\mathbf{x}}) \quad \textit{s.t.} \quad y_i = y,\ \mathbf{h}_i(\mathbf{x}) \geq \mathbf{0},\ \mathbf{c}(\mathbf{x}) = \mathbf{0},\ \mathbf{d}(\mathbf{x}) \geq \mathbf{0}.$$

- Solving problem (1) is equivalent to solving it within each leaf's region and then picking the leaf with the best solution.

Counterfactual explanations in oblique trees

In an oblique decision tree each leaf region is an polytope defined by intersection of arbitrary hyperplanes found in the path from root to leaf.

- For each target leaf the problem becomes:

$$\min_{\mathbf{x}\in\mathbb{R}^D} E(\mathbf{x};\overline{\mathbf{x}}) \quad \text{s.t.} \quad y_i = y,\ \mathbf{h}_i(\mathbf{x}) \geq \mathbf{0},\ \mathbf{c}(\mathbf{x}) = \mathbf{0},\ \mathbf{d}(\mathbf{x}) \geq \mathbf{0}.$$

- $\mathbf{h}_i(\mathbf{x})$ is the set of hyperplanes that represents decision rule of the nodes in the path from root to leaf i.
- $\mathbf{h}_i(\mathbf{x})$ forms set of linear constraints.
- If E is ℓ_2 or ℓ_1 distance, then the problem becomes QP or LP, which can be solved very effectively.

Counterfactual explanations in axis-aligned trees

In an axis-aligned decision tree each leaf region is an axis-aligned boxes defined by intersection of axis-aligned hyperplanes found in the path from root to leaf.

Theorem

In problem (1)*, assume that each constraint depends on a single element of* $\mathbf{x}$ *(not necessarily the same) and that the objective function is separable, i.e.,* $E(\mathbf{x};\overline{\mathbf{x}}) = \sum_{d=1}^{D} E_d(x_d;\overline{x}_d)$*. Then the problem separates over the variables* $x_1, \dots, x_D$*.*

- This applies to axis-aligned trees because each of the constraints $\mathbf{h}_i(\mathbf{x}) \geq \mathbf{0}$ in the path from the root to leaf i involve a single feature of $\mathbf{x}$ (they are bound constraints).
- This means that, in axis-aligned tree within each leaf, we can solve for each x_d independently, by minimizing $E_d(x_d;\overline{x}_d)$ subject to the constraints on x_d.

Counterfactual explanations in axis-aligned trees cont.

Theorem

Consider the scalar constrained optimization problem, where the bounds can take the values $l_d = -\infty$ and $u_d = \infty$:

$$\min_{x_d \in \mathbb{R}} E_d(x_d; \overline{x}_d) \quad \textit{s.t.} \quad l_d \leq x_d \leq u_d.$$

Assume E_d is convex on x_d and satisfies $E_d(\overline{x}_d; \overline{x}_d) = 0$ and $E_d(x_d; \overline{x}_d) \geq 0$ $\forall x_d \in \mathbb{R}$. Then x_d^*, defined as the median of $\overline{x}_d$, l_d and u_d, is a global minimizer of the problem:

$$x_d^* = \textit{median}(\overline{x}_d, l_d, u_d) = \begin{cases} l_d, & \overline{x}_d < l_d \\ u_d, & \overline{x}_d > u_d \\ \overline{x}_d, & \textit{otherwise} \end{cases}.$$

- This makes solving the counterfactual explanation problem exceedingly fast for axis-aligned trees.

Exploring different types of counterfactual explanation questions

Finding the closest boundary. The minimum-distance change to $\overline{\mathbf{x}}$ that changes its original class k.

- Find the counterfactual explanation in every leaf except the ones with label k.
- Pick the counterfactual with the lowest cost.

Exploring different types of counterfactual explanation questions

Critical attribute for change to the target class y. Which attribute has the lowest cost to change the class of $\overline{\mathbf{x}}$ to a target class y, if changing only one attribute?

- For given a attribute d, add all other attributes to the equality constraint $(\mathbf{c}(\mathbf{x}) = \mathbf{0})$ and solve the counterfactual problem as described in previous slides.
- Repeat above step for each attribute in $\overline{\mathbf{x}}$.
- Pick the attribute for which the counterfactual $(\mathbf{x}^*)$ has the lowest cost.

Exploring different types of counterfactual explanation questions

Critical attribute for changing the class. Which attribute has the lowest cost to change the class of $\overline{\mathbf{x}}$ to any other class if changing only one attribute?

- For given a attribute d, add all other attributes to the equality constraint $(\mathbf{c}(\mathbf{x}) = \mathbf{0})$ and find the closest boundary.
- Repeat above step for each attribute in $\overline{\mathbf{x}}$.
- Pick the attribute for which the counterfactual $(\mathbf{x}^*)$ has the lowest cost.

Use case study 1a

Multiclass classification {excellent, good, satisfactory, sufficient or fail}.

Attribute	$\overline{\mathbf{x}}$, source instance		
school	GP		
sex	male		
age	18		
parent's status	together		
mother's job	services		
father's job	services		
guardian	mother		
previous class failures	2		
school support	no		
family support	no		
study time[◇]	1		
plan for higher education	no		
internet access	yes		
family relationship[‡]	3		
free time[‡]	2		
going out frequency[‡]	5		
health[‡]	5		
absences[‡]	4		
Grades	fail		

[◇] 1– <15 min, 2– 15 to 30 min., 3– 30 min. to1 hour or 4 – > 1 hour). [‡] from 1 = very low to 5 = very high.

Use case study 1a

Multiclass classification {excellent, good, satisfactory, sufficient or fail}.

Attribute	$\overline{\mathbf{x}}$, source instance	$\mathbf{x}_1^*$, target class excellent	
school	GP	=	
sex	male	=	
age	18	=	
parent’s status	together	=	
mother’s job	services	teacher	
father’s job	services	teacher	
guardian	mother	=	
previous class failures	2	1	
school support	no	=	
family support	no	=	
study time$^\diamond$	1	=	
plan for higher education	no	yes	
internet access	yes	=	
family relationship‡	3	=	
free time‡	2	=	
going out frequency‡	5	=	
health‡	5	=	
absences‡	4	=	
Grades	fail	excellent	

$^\diamond$ 1– <15 min, 2– 15 to 30 min., 3– 30 min. to1 hour or 4 – > 1 hour). ‡ from 1 = very low to 5 = very high.

Use case study 1a

Multiclass classification {excellent, good, satisfactory, sufficient or fail}.

Attribute	$\overline{\mathbf{x}}$, source instance	$\mathbf{x}_1^*$, target class excellent	$\mathbf{x}_2^*$, closest class
school	GP	=	=
sex	male	=	=
age	18	=	=
parent's status	together	=	=
mother's job	services	teacher	=
father's job	services	teacher	=
guardian	mother	=	=
previous class failures	2	1	1
school support	no	=	=
family support	no	=	=
study time$^\diamond$	1	=	=
plan for higher education	no	yes	=
internet access	yes	=	=
family relationship‡	3	=	=
free time‡	2	=	=
going out frequency‡	5	=	=
health‡	5	=	=
absences‡	4	=	=
Grades	fail	excellent	sufficient

$^\diamond$ 1– <15 min, 2– 15 to 30 min., 3– 30 min. to1 hour or 4 – > 1 hour). ‡ from 1 = very low to 5 = very high.

Use case study 1b

Multiclass classification {excellent, good, satisfactory, sufficient or fail}.

Attribute	$\overline{\mathbf{x}}$, source instance		
school	MS		
sex	male		
age	17		
parent's status	together		
mother's job	other		
father's job	other		
guardian	mother		
previous class failures	0		
school support	no		
family support	no		
study time$^{\diamond}$	2		
plan for higher education	yes		
internet access	yes		
family relationship‡	4		
free time‡	4		
going out frequency‡	3		
health‡	5		
absences	4		
Grades	fail		

$^{\diamond}$ 1– $<$15 min, 2– 15 to 30 min., 3– 30 min. to1 hour or 4 – $>$ 1 hour). ‡ from 1 = very low to 5 = very high.

Use case study 1b

Multiclass classification {excellent, good, satisfactory, sufficient or fail}.

Attribute	$\overline{\mathbf{x}}$, source instance	$\mathbf{x}_1^*$, target class satisfactory
school	MS	=
sex	male	=
age	17	=
parent's status	together	=
mother's job	other	=
father's job	other	=
guardian	mother	=
previous class failures	0	=
school support	no	=
family support	no	=
study time$^\diamond$	2	=
plan for higher education	yes	=
internet access	yes	=
family relationship‡	4	=
free time‡	4	1
going out frequency‡	3	=
health‡	5	=
absences	4	=
Grades	fail	satisfactory

$^\diamond$ 1– <15 min, 2– 15 to 30 min., 3– 30 min. to1 hour or 4 – > 1 hour). ‡ from 1 = very low to 5 = very high.

Use case study 1b

Multiclass classification {excellent, good, satisfactory, sufficient or fail}.

Attribute	$\overline{\mathbf{x}}$, source instance	$\mathbf{x}_1^*$, target class satisfactory	$\mathbf{x}_2^*$, closest class
school	MS	=	=
sex	male	=	=
age	17	=	=
parent's status	together	=	=
mother's job	other	=	=
father's job	other	=	=
guardian	mother	=	=
previous class failures	0	=	=
school support	no	=	=
family support	no	=	=
study time$^\diamond$	2	=	=
plan for higher education	yes	=	=
internet access	yes	=	=
family relationship‡	4	=	=
free time‡	4	1	=
going out frequency‡	3	=	2
health‡	5	=	=
absences	4	=	=
Grades	fail	satisfactory	sufficient

$^\diamond$ 1– $<$15 min, 2– 15 to 30 min., 3– 30 min. to1 hour or 4 – $>$ 1 hour). ‡ from 1 = very low to 5 = very high.

Exploring different types of counterfactual explanation questions

Robust counterfactuals. Here, we want to find the counterfactuals that are well inside a leaf region rather than on the boundary, so they are more robust to flipping their class due to small changes.

- This problem can easily be solved by shrinking the leaf region size. That is for solving the counterfactual problem in a leaf region, the constraint "$\mathbf{h}_i(\mathbf{x}) \geq \mathbf{0}$" becomes "$\mathbf{h}_i(\mathbf{x}) \geq \epsilon$", where $\epsilon > 0$.

Use case study 2

Binary classification {bad credit and good credit}. $\overline{\mathbf{x}}$ is classified as bad credit (using a pre-trained tree), and target class is good credit.

Attribute	$\overline{\mathbf{x}}$, source instance				
existing checking	$<$ 0 DM				
duration	15 months				
credithistory	critical account				
purpose	furniture/equipment				
credit amount	1478				
savings	$<$ 100 DM				
employment since	$\geq$ 7 years				
iInstallment rate	4				
status and sex	male : single				
other debtors	none				
residence since	4				
property	car or other				
age	44				
other installment plans	none				
housing	own				
existing credits	2				
job	skilled employee				
people liable	2				
telephone	yes				
foreignworker	yes				
Credit	Bad				

Use case study 2

Binary classification {bad credit and good credit}. $\overline{\mathbf{x}}$ is classified as bad credit (using a pre-trained tree), and target class is good credit.

Attribute	$\overline{\mathbf{x}}$, source instance	$\mathbf{x}_1^*$, $\epsilon = 0.00$ $\ell_2 = 1.73$		
existing checking	$<$ 0 DM	$=$		
duration	15 months	16 months		
credithistory	critical account	$=$		
purpose	furniture/ equipment	$=$		
credit amount	1478	$=$		
savings	$<$ 100 DM	$=$		
employment since	$\geq$ 7 years	$=$		
iInstallment rate	4	$=$		
status and sex	male : single	$=$		
other debtors	none	$=$		
residence since	4	$=$		
property	car or other	$=$		
age	44	$=$		
other installment plans	none	stores		
housing	own	$=$		
existing credits	2	$=$		
job	skilled employee	$=$		
people liable	2	$=$		
telephone	yes	$=$		
foreignworker	yes	$=$		
Credit	Bad	Good		

Use case study 2

Binary classification {bad credit and good credit}. $\overline{\mathbf{x}}$ is classified as bad credit (using a pre-trained tree), and target class is good credit.

Attribute	$\overline{\mathbf{x}}$, source instance	$\mathbf{x}_1^*$, $\epsilon = 0.00$ $\ell_2 = 1.73$	$\mathbf{x}_2^*$, $\epsilon = 0.10$ $\ell_2 = 1.73$	$\mathbf{x}_3^*$, $\epsilon = 0.20$ $\ell_2 = 2.00$	$\mathbf{x}_4^*$, $\epsilon = 0.25$ $\ell_2 = 2.23$
existing checking	$<$ 0 DM	=	=	=	=
duration	15 months	16 months	=	=	16 months
credithistory	critical account	=	=	delay in paying off in the past	delay in paying off in the past
purpose	furniture/ equipment	=	=	=	=
credit amount	1478	=	=	=	=
savings	$<$ 100 DM	=	=	=	=
employment since	$\geq$ 7 years	=	=	=	=
iInstallment rate	4	=	=	=	=
status and sex	male : single	=	=	=	=
other debtors	none	=	=	=	=
residence since	4	=	3	3	3
property	car or other	=	=	=	=
age	44	=	=	=	=
other installment plans	none	stores	stores	=	=
housing	own	=	=	=	=
existing credits	2	=	=	1	1
job	skilled employee	=	=	=	=
people liable	2	=	=	=	=
telephone	yes	=	=	=	=
foreignworker	yes	=	=	=	=
Credit	Bad	Good	Good	Good	Good

Regression Trees

- We can also use our approach to explore more practical problems that are related to the regression trees.
- Consider a regression tree T:
  - $T(\overline{\mathbf{x}})$ is the predicted value of the source instance $(\overline{\mathbf{x}})$.
  - $T(\mathbf{x}^*)$ represents the predicted value of the counterfactual $(\mathbf{x}^*)$.

Exploring different types of counterfactual explanation questions

$T(\mathbf{x}^*) > T(\overline{\mathbf{x}})$: find the minimum change in $\overline{\mathbf{x}}$ that increase its predicted value.

- Consider the leaves whose label is larger than the $T(\overline{\mathbf{x}})$ as target leaves.
- Find counterfactual $(\mathbf{x}^*)$ in each target leaf.
- Pick the $\mathbf{x}^*$ with the lowest cost.

Exploring different types of counterfactual explanation questions

$T(\mathbf{x}^*) \geq T(\overline{\mathbf{x}}) + \beta$: find the minimum change in $\overline{\mathbf{x}}$ that increase its predicted value atleast by β.

- Consider the leaves whose label is larger than or equal to $T(\overline{\mathbf{x}}) + \beta$ as target leaves.
- Find counterfactual $(\mathbf{x}^*)$ in each target leaf.
- Pick the $\mathbf{x}^*$ with the lowest cost.

Exploring different types of counterfactual explanation questions

$\alpha \geq T(\mathbf{x}^*) \geq \beta$: find the minimum change in $\overline{\mathbf{x}}$ that change its predicted value between α and β.

- Consider the leaves with label between α and β.
- Find counterfactual $(\mathbf{x}^*)$ in each target leaf.
- Pick the $\mathbf{x}^*$ with the lowest cost.

Use case study 3

Regression task to predict Median home value.

Attribute	$\overline{x}$, source instance			
crime rate	2.37			
residential land zoned proportion	0.0			
proportion of non-retail business	19.58			
tract bounds river	0			
nitric oxides concentration	0.87			
avg. rooms per dwelling	4.92			
proportion of units before 1940	95.70			
distances to Boston employment centres	1.46			
accessibility to highways	5.00			
property-tax rate	403.00			
pupil-teacher ratio	14.70			
proportion of african american by town	391.71			
% lower status of the population	29.53			
Median home value in $1000's	14.74			

Use case study 3

Regression task to predict Median home value.

Attribute	$\overline{x}$, source instance	$T(\mathbf{x}^*) >$ $T(\overline{\mathbf{x}})$	
crime rate	2.37	2.15	
residential land zoned proportion	0.0	0.02	
proportion of non-retail business	19.58	19.48	
tract bounds river	0	=	
nitric oxides concentration	0.87	0.39	
avg. rooms per dwelling	4.92	5.13	
proportion of units before 1940	95.70	95.67	
distances to Boston employment centres	1.46	1.17	
accessibility to highways	5.00	5.04	
property-tax rate	403.00	=	
pupil-teacher ratio	14.70	14.47	
proportion of african american by town	391.71	=	
% lower status of the population	29.53	29.39	
Median home value in $1000's	14.74	15.96	

Use case study 3

Regression task to predict Median home value.

Attribute	$\overline{x}$, source instance	$T(\mathbf{x}^*) > T(\overline{\mathbf{x}})$	$T(\mathbf{x}^*) \geq T(\overline{\mathbf{x}}) + 5$
crime rate	2.37	2.15	1.93
residential land zoned proportion	0.0	0.02	=
proportion of non-retail business	19.58	19.48	=
tract bounds river	0	=	1
nitric oxides concentration	0.87	0.39	0.39
avg. rooms per dwelling	4.92	5.13	5.73
proportion of units before 1940	95.70	95.67	95.71
distances to Boston employment centres	1.46	1.17	=
accessibility to highways	5.00	5.04	5.1
property-tax rate	403.00	=	=
pupil-teacher ratio	14.70	14.47	14.58
proportion of african american by town	391.71	=	=
% lower status of the population	29.53	29.39	29.54
Median home value in $1000's	14.74	15.96	20.52

Use case study 3

Regression task to predict Median home value.

Attribute	$\overline{x}$, source instance	$T(\mathbf{x}^*) > T(\overline{\mathbf{x}})$	$T(\mathbf{x}^*) \geq T(\overline{\mathbf{x}}) + 5$	$25 \geq T(\mathbf{x}^*) \geq 30$
crime rate	2.37	2.15	1.93	1.81
residential land zoned proportion	0.0	0.02	=	0.03
proportion of non-retail business	19.58	19.48	=	19.48
tract bounds river	0	=	1	=
nitric oxides concentration	0.87	0.39	0.39	0.385
avg. rooms per dwelling	4.92	5.13	5.73	8.09
proportion of units before 1940	95.70	95.67	95.71	95.66
distances to Boston employment centres	1.46	1.17	=	1.16
accessibility to highways	5.00	5.04	5.1	5.41
property-tax rate	403.00	=	=	402.99
pupil-teacher ratio	14.70	14.47	14.58	14.60
proportion of african american by town	391.71	=	=	391.67
% lower status of the population	29.53	29.39	29.54	29.41
Median home value in $1000's	14.74	15.96	20.52	29.14

Conclusion

- Classification and regression trees are important, particularly in applications where interpretability is desirable, such as business, law, and medicine.
- Sparse Oblique decision trees, trained by the TAO algorithm, can be surprisingly accurate and interprtable.
- The counterfactual explanation problem for classification trees (axis-aligned and oblique) is nonconvex and nondifferentiable but can be solved exactly and efficiently.
- Proposed approach can handle several useful distance functions and linear constraints (equality and inequality); and is applicable to both continuous and categorical variables.
- The formulation can be applied to answer a variety of practical questions and is fast enough for interactive use.