# Nearest neighbor for realizable online classification

Sanjoy Dasgupta and Geelon So (2023)

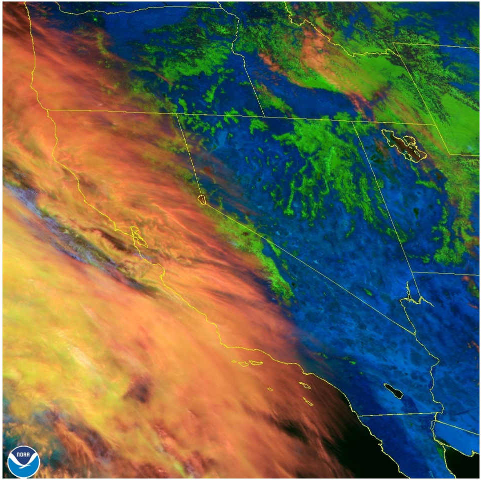
Geelon So, agso@eng.ucsd.edu
EnCORE Student Social — Mar 20, 2023

# Weather forecasting problem

**The Weather Channel's task**

Each day:

- receive atmospheric data

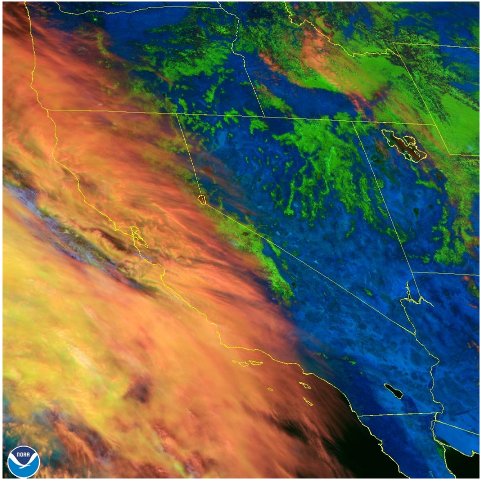


# Weather forecasting problem

### THE WEATHER CHANNEL'S TASK

Each day:

- receive atmospheric data
- predict tomorrow's weather

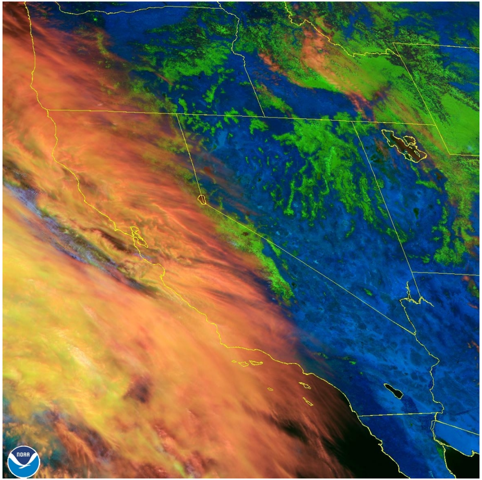


# Weather forecasting problem

**THE WEATHER CHANNEL'S TASK**

Each day:

- ▶ receive atmospheric data
- ▶ predict tomorrow's weather
- ▶ observe actual weather

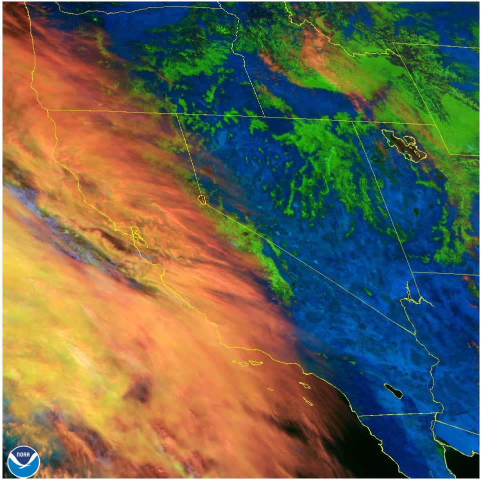


# Weather forecasting problem

THE WEATHER CHANNEL'S TASK

Each day:

- receive atmospheric data
- predict tomorrow's weather
- observe actual weather
- incur ire of viewers if wrong



# Nearest neighbor for weather prediction

NEAREST NEIGHBOR ALGORITHM

- remember all past conditions + weather outcomes

# Nearest neighbor for weather prediction

**NEAREST NEIGHBOR ALGORITHM**

- remember all past conditions + weather outcomes
- predict weather according to the most similar conditions in memory

# The nearest neighbor rule

**SETTING**

Let $(\mathcal{X}, \rho)$ be a metric space.

# The nearest neighbor rule

**SETTING**

Let $(\mathcal{X}, \rho)$ be a metric space.

**NEAREST NEIGHBOR ALGORITHM**

# The nearest neighbor rule

**SETTING**

Let $(\mathcal{X}, \rho)$ be a metric space.

**NEAREST NEIGHBOR ALGORITHM**

- remember all past data points $\{(x_1, y_1), \ldots, (x_t, y_t)\}$

# The nearest neighbor rule

**SETTING**

Let $(\mathcal{X}, \rho)$ be a metric space.

**NEAREST NEIGHBOR ALGORITHM**

- remember all past data points $\{(x_1, y_1), \ldots, (x_t, y_t)\}$
- given query $x$, find most similar data point in memory

$$\mathrm{NN}(x) = \arg\min_{\tau} \rho(x, x_\tau)$$

# The nearest neighbor rule

**SETTING**

Let $(\mathcal{X}, \rho)$ be a metric space.

**NEAREST NEIGHBOR ALGORITHM**

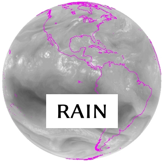
- remember all past data points $\{(x_1, y_1), \ldots, (x_t, y_t)\}$
- given query $x$, find most similar data point in memory

$$\mathrm{NN}(x) = \arg\min_{\tau} \rho(x, x_\tau)$$

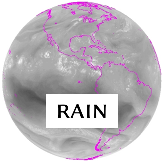
- predict using corresponding label

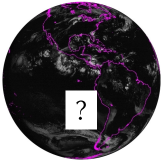
$$\hat{y}(x) = y_{\mathrm{NN}(x)}$$

# Nearest neighbor for weather prediction

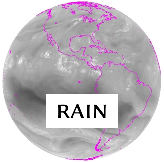


# Nearest neighbor for weather prediction

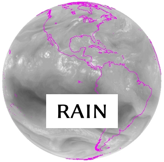


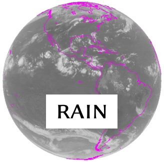



# Nearest neighbor for weather prediction

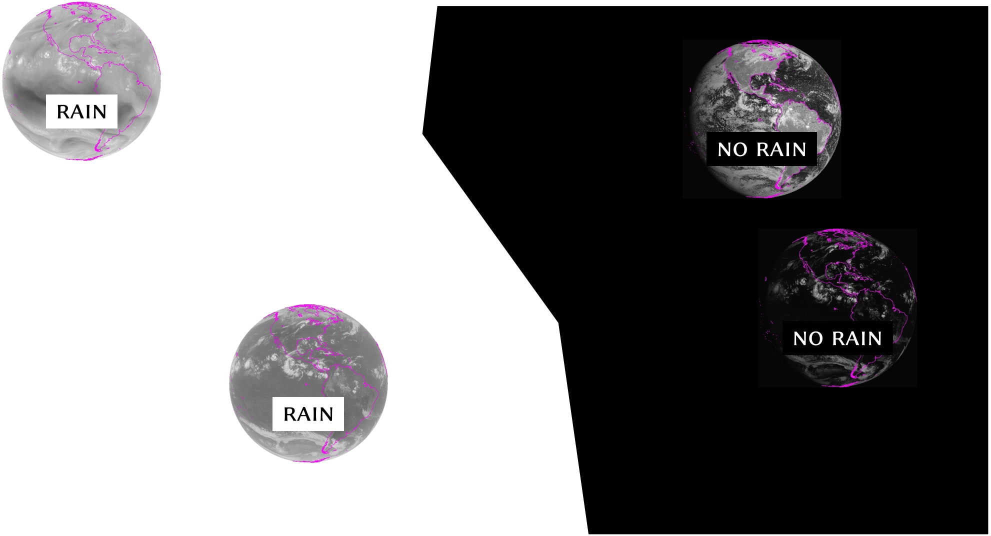



# Nearest neighbor for weather prediction

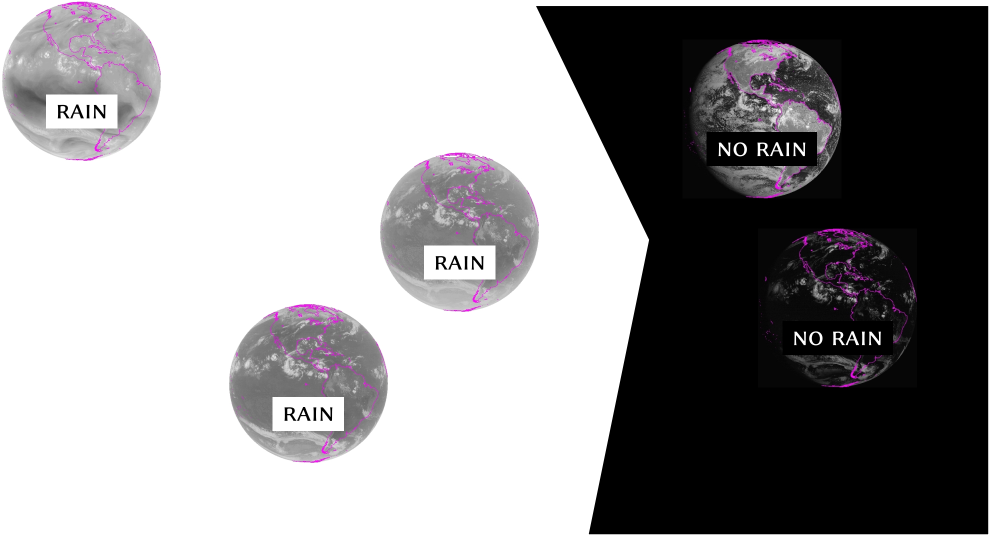


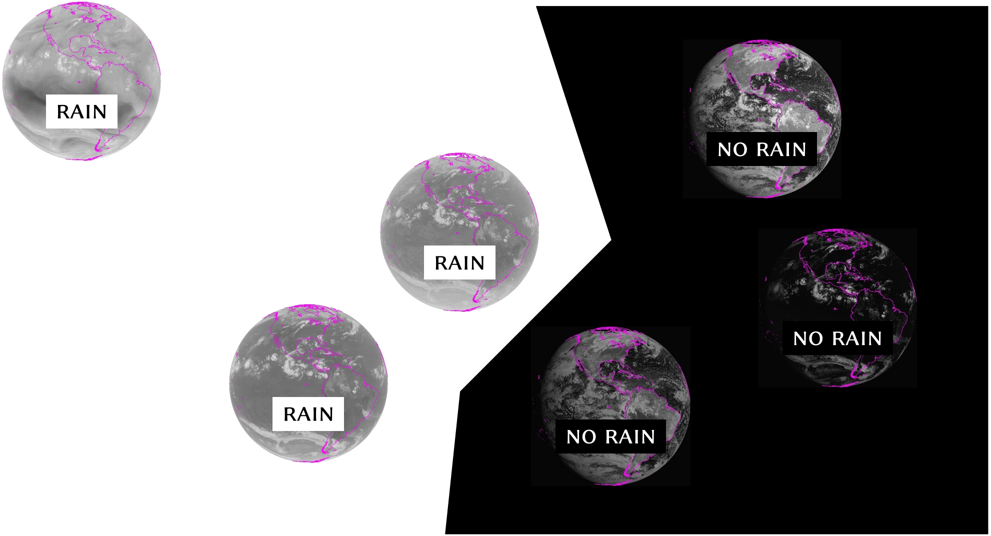

# Nearest neighbor for weather prediction



# Nearest neighbor for weather prediction



# Nearest neighbor for weather prediction



# Behavior of online nearest neighbor

**QUESTION**

When is the *nearest neighbor rule* a reasonable online prediction strategy?

# Online learning setting

**ONLINE LEARNING LOOP**

For $t = 1, 2, \ldots$

- receive instance $x_t$

# Online learning setting

**ONLINE LEARNING LOOP**

For $t = 1, 2, \ldots$

- receive instance $x_t$
- predict label $\hat{y}_t$

# Online learning setting

**ONLINE LEARNING LOOP**

For $t = 1, 2, \ldots$

- receive instance $x_t$
- predict label $\hat{y}_t$
- observe true label $y_t$

# Online learning setting

**ONLINE LEARNING LOOP**

For $t = 1, 2, \ldots$

- receive instance $x_t$
- predict label $\hat{y}_t$
- observe true label $y_t$
- incur loss $\ell(x_t, y_t, \hat{y}_t)$

# Online learning setting

REALIZABILITY ASSUMPTION

The true labels are generated by some underlying function $f : \mathcal{X} \to \mathcal{Y}$,

$$y_t = f(x_t).$$

# Online learning setting

**GOAL**

Make fewer and fewer mistakes over time.

# Online learning setting

**GOAL**

Make fewer and fewer mistakes over time. Formally:

$$\underbrace{\mathrm{er}_T := \frac{1}{T}\sum_{t=1}^{T} \ell\big(x_t, y_t, \hat{y}_t\big) \to 0}_{\text{achieve vanishing error rate}} .$$

# Connection to regret

In the usual goal in the online learning setting is to achieve sublinear regret:

$$\text{regret}_T := \sum_{t=1}^{T} \ell\big(x_t, y_t, \hat{y}_t\big) - \inf_{h \in \mathcal{H}} \sum_{t=1}^{T} \ell\big(x_t, y_t, h(x_t)\big).$$

# Connection to regret

In the usual goal in the online learning setting is to achieve sublinear regret:

$$\mathrm{regret}_T := \sum_{t=1}^{T} \ell\big(x_t, y_t, \hat{y}_t\big) - \inf_{h\in\mathcal{H}} \sum_{t=1}^{T} \ell\big(x_t, y_t, h(x_t)\big).$$

- In the realizable setting, if $\mathcal{H}$ is non-parametric (e.g. all nearest neighbor classifiers), no mistakes are made by any optimal $h \in \mathcal{H}$ on $(x_1, y_1), \ldots, (x_T, y_T)$.

# Connection to regret

In the usual goal in the online learning setting is to achieve sublinear regret:

$$\text{regret}_T := \sum_{t=1}^{T} \ell\big(x_t, y_t, \hat{y}_t\big) - \inf_{h \in \mathcal{H}} \sum_{t=1}^{T} \ell\big(x_t, y_t, h(x_t)\big).$$

- In the realizable setting, if $\mathcal{H}$ is non-parametric (e.g. all nearest neighbor classifiers), no mistakes are made by any optimal $h \in \mathcal{H}$ on $(x_1, y_1), \ldots, (x_T, y_T)$.
- Thus, sublinear regret is equivalent to vanishing error rate.

# Difficulty of realizable online learning

- The sequence of instances $x_t$ do not come i.i.d. from some distribution.

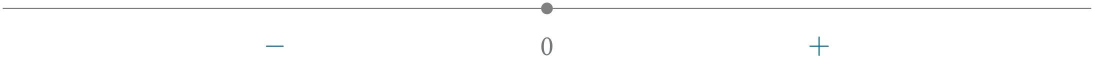
# Difficulty of realizable online learning

- The sequence of instances $x_t$ do not come i.i.d. from some distribution.
- In the worst-case, each $x_t$ is selected so that learner makes a mistake each time.

# Negative example: learning the sign function

**GOAL**

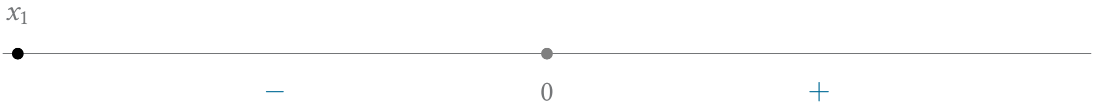
Learn the sign function $f(x) := \begin{cases} + & x \geq 0 \\ - & x < 0 \end{cases}$



# Negative example: learning the sign function

**GOAL**

Learn the sign function $\quad f(x) := \begin{cases} + & x \geq 0 \\ - & x < 0 \end{cases}$

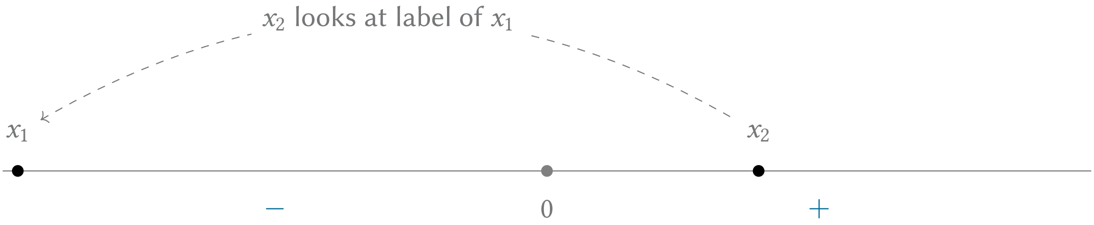



**EXAMPLE.** A worst-case sequence where the nearest neighbor rule errs every time.

# Negative example: learning the sign function

**GOAL**

Learn the sign function $f(x) := \begin{cases} + & x \geq 0 \\ - & x < 0 \end{cases}$

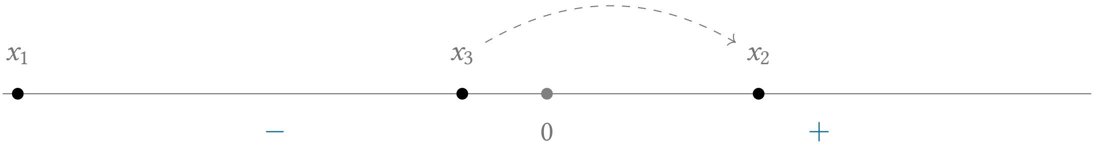



**EXAMPLE.** A worst-case sequence where the nearest neighbor rule errs every time.

# Negative example: learning the sign function

**GOAL**

Learn the sign function $\quad f(x) := \begin{cases} + & x \geq 0 \\ - & x < 0 \end{cases}$

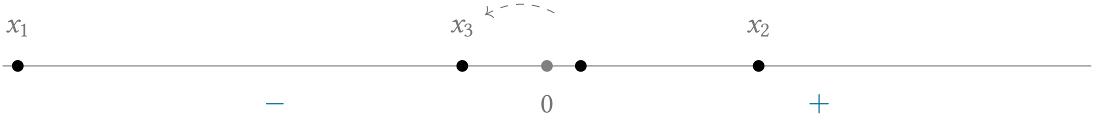



**EXAMPLE.** A worst-case sequence where the nearest neighbor rule errs every time.

# Negative example: learning the sign function

**GOAL**

Learn the sign function $f(x) := \begin{cases} + & x \geq 0 \\ - & x < 0 \end{cases}$




**EXAMPLE.** A worst-case sequence where the nearest neighbor rule errs every time.

- The sequence alternate signs and the nearest neighbor of $x_{t+1}$ is $x_t$ out of $x_1, \ldots, x_t$.

# Negative result

**SETTING**

Let $(\mathcal{X}, \rho)$ be a totally bounded metric space and $f : \mathcal{X} \to \{-, +\}$.

# Negative result

**SETTING**
Let $(\mathcal{X}, \rho)$ be a totally bounded metric space and $f : \mathcal{X} \to \{-, +\}$.

## Proposition (Non-convergence in the worst-case)

*There is a sequence of instances $(x_t)_t$ on which the nearest neighbor error rate is bounded away from zero if and only if there is no positive separation between classes:*

$$\inf_{f(x) \neq f(x')} \rho(x, x') = 0.$$

# Negative result

**SETTING**

Let $(\mathcal{X}, \rho)$ be a totally bounded metric space and $f : \mathcal{X} \to \{-, +\}$.

## Proposition (Non-convergence in the worst-case)

*There is a sequence of instances $(x_t)_t$ on which the nearest neighbor **error rate is bounded away from zero** if and only if there is **no positive separation between classes**:*

$$\inf_{f(x) \neq f(x')} \rho(x, x') = 0.$$

- **Proof idea:** can always find arbitrarily close pairs $(x, x')$ with opposite signs
  - can select sequence so that $x_{2t}$ is closest to $x_{2t-1}$, which has the opposite sign

# Implications of negative result

The worst-case adversary is too powerful—learning may not be possible in this setting.

- ISSUE: the adversary can compute/construct test instances with arbitrary precision.

# Implications of negative result

The worst-case adversary is too powerful—learning may not be possible in this setting.

- **ISSUE:** the adversary can compute/construct test instances with arbitrary precision.

Saving grace: we may reasonably never expect to meet this worst-case adversary.

# Implications of negative result

The worst-case adversary is too powerful—learning may not be possible in this setting.

- ISSUE: the adversary can compute/construct test instances with arbitrary precision.

Saving grace: we may reasonably never expect to meet this worst-case adversary.

- a real adversary may be limited in:
  - information,

# Implications of negative result

The worst-case adversary is too powerful—learning may not be possible in this setting.

- ISSUE: the adversary can compute/construct test instances with arbitrary precision.

Saving grace: we may reasonably never expect to meet this worst-case adversary.

- a real adversary may be limited in:
  - information,
  - computational power,

# Implications of negative result

The worst-case adversary is too powerful—learning may not be possible in this setting.

- ISSUE: the adversary can compute/construct test instances with arbitrary precision.

Saving grace: we may reasonably never expect to meet this worst-case adversary.

- a real adversary may be limited in:
    - information,
    - computational power,
    - access to hard instances.

# This work

RESEARCH QUESTION

Under what *general conditions* is realizable online learning possible?

# This work

**RESEARCH QUESTION**

Under what *general conditions* is realizable online learning possible?

- How much do we need to relax the worst-case adversary?

# Smoothed analysis of algorithms

Spielman and Teng (2004) initiated the analysis of algorithms beyond the worst-case.

# Smoothed analysis of algorithms

Spielman and Teng (2004) initiated the analysis of algorithms beyond the worst-case.

- Worst-case analysis:
  - Show that there exists at least one hard problem instance.

# Smoothed analysis of algorithms

Spielman and Teng (2004) initiated the analysis of algorithms beyond the worst-case.

- Worst-case analysis:
  - Show that there exists at least one hard problem instance.
  - This can fail to capture the actual behavior of the algorithm in practice.

# Smoothed analysis of algorithms

Spielman and Teng (2004) initiated the analysis of algorithms beyond the worst-case.

- Worst-case analysis:
  - Show that there exists at least one hard problem instance.
  - This can fail to capture the actual behavior of the algorithm in practice.
- Non-worst-case analysis:
  - Introduce a (probability) measure over problem instances.

# Smoothed analysis of algorithms

Spielman and Teng (2004) initiated the analysis of algorithms beyond the worst-case.

- Worst-case analysis:
  - Show that there exists at least one hard problem instance.
  - This can fail to capture the actual behavior of the algorithm in practice.
- Non-worst-case analysis:
  - Introduce a (probability) measure over problem instances.
  - Show that almost all problems are easy (the hard instances have measure zero).
    - Or, problems are easy with high probability/on average.

# Smoothed adversary for online learning

**SMOOTHED ADVERSARY LOOP**

For $t = 1, 2, \ldots$

# Smoothed adversary for online learning

SMOOTHED ADVERSARY LOOP

For $t = 1, 2, \ldots$

- select test distribution $\mu_t$ over all instances $\mathcal{X}$

# Smoothed adversary for online learning

SMOOTHED ADVERSARY LOOP

For $t = 1, 2, \ldots$

- **select** test distribution $\mu_t$ over all instances $\mathcal{X}$
- **draw** test instance $x_t \sim \mu_t$

# Smoothed adversary for online learning

SMOOTHED ADVERSARY LOOP

For $t = 1, 2, \ldots$

- select test distribution $\mu_t$ over all instances $\mathcal{X}$
- draw test instance $x_t \sim \mu_t$
- observe (fully/partially) learner's internal state after update

# Smoothed adversary for online learning

**SMOOTHED ADVERSARY LOOP**

For $t = 1, 2, \ldots$

- select test distribution $\mu_t$ over all instances $\mathcal{X}$
- draw test instance $x_t \sim \mu_t$
- observe (fully/partially) learner's internal state after update

The smoothed online setting is also studied by Rakhlin et al. (2011); Haghtalab et al. (2020).

# Smoothed adversary for online learning

- The smoothed adversary framework interpolates between:
  - the i.i.d. setting: $\mu_t$ is fixed for all time $t$

# Smoothed adversary for online learning

- The smoothed adversary framework interpolates between:
  - the i.i.d. setting: $\mu_t$ is fixed for all time $t$
  - the worst-case setting: $\mu_t$ may be point masses

# Example: Gaussian perturbation model

**GAUSSIAN-SMOOTHED ADVERSARY:**

- adversary selects $\overline{x}$

# Example: Gaussian perturbation model

**GAUSSIAN-SMOOTHED ADVERSARY:**

- adversary selects $\overline{x}$
- test instance $x$ is a perturbed version $\overline{x} + \xi$ where $\xi \sim \mathcal{N}(0, \sigma^2 I)$, so:

$$\mu = \mathcal{N}(\bar{x}, \sigma^2 I).$$

# Example: $\sigma$-smoothed adversary

$\sigma$-SMOOTHED ADVERSARY:

- let $\nu$ be an underlying distribution over $\mathcal{X}$

# Example: $\sigma$-smoothed adversary

**$\sigma$-SMOOTHED ADVERSARY:**

- let $\nu$ be an underlying distribution over $\mathcal{X}$
- the adversary can select any distribution $\mu$ satisfying:

$$\mu(A) \leq \frac{1}{\sigma} \cdot \nu(A),$$

for all $A \subset \mathcal{X}$ measurable.

# Dominated adversary

In this work, we generalize both by the $\nu$-*dominated adversary*.

- **INTUITION:** the dominated adversary cannot place a constant probability mass on an arbitrarily small region of problem instances.

# Dominated adversary

In this work, we generalize both by the $\nu$-*dominated adversary*.

- INTUITION: the dominated adversary cannot place a constant probability mass on an arbitrarily small region of problem instances.
- SETTING: let $(\mathcal{X}, \nu)$ be a measure space.

# Dominated adversary

In this work, we generalize both by the $\nu$*-dominated adversary*.

- **INTUITION:** the dominated adversary cannot place a constant probability mass on an arbitrarily small region of problem instances.
- **SETTING:** let $(\mathcal{X}, \nu)$ be a measure space.

## Definition (Dominated adversary)

*The measure $\nu$ uniformly dominates a family $\mathcal{M}$ of probability distributions on $\mathcal{X}$ if for all $\varepsilon > 0$ there exists $\delta > 0$ such that:*

$$\nu(A) < \delta \quad \Longrightarrow \quad \mu(A) < \varepsilon,$$

*for all $A \subset \mathcal{X}$ measurable and distribution $\mu \in \mathcal{M}$.*

# Dominated adversary

In this work, we generalize both by the $\nu$*-dominated adversary*.

- **INTUITION:** the dominated adversary cannot place a constant probability mass on an arbitrarily small region of problem instances.
- **SETTING:** let $(\mathcal{X}, \nu)$ be a measure space.

## Definition (Dominated adversary)

*The measure $\nu$ uniformly dominates a family $\mathcal{M}$ of probability distributions on $\mathcal{X}$ if for all $\varepsilon > 0$ there exists $\delta > 0$ such that:*

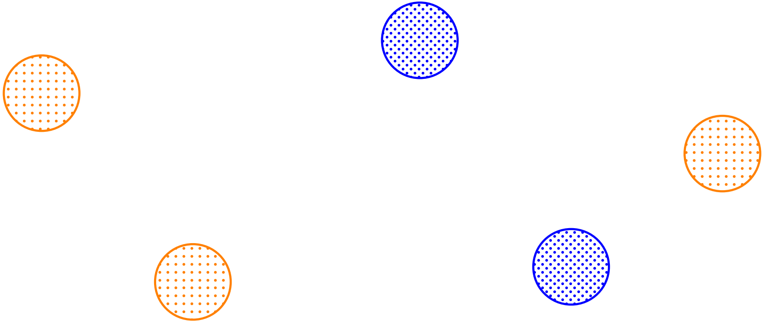
$$\nu(A) < \delta \quad \Longrightarrow \quad \mu(A) < \varepsilon,$$

*for all $A \subset \mathcal{X}$ measurable and distribution $\mu \in \mathcal{M}$. We say that adversary is $\nu$-dominated if at all times $t$ it selects $\mu_t$ from a family of distributions uniformly dominated by $\nu$.*

# Example: learning labels for well-separated clusters

**SETTING**

Suppose that the instance space $\mathcal{X}$ consists of countably many well-separated clusters

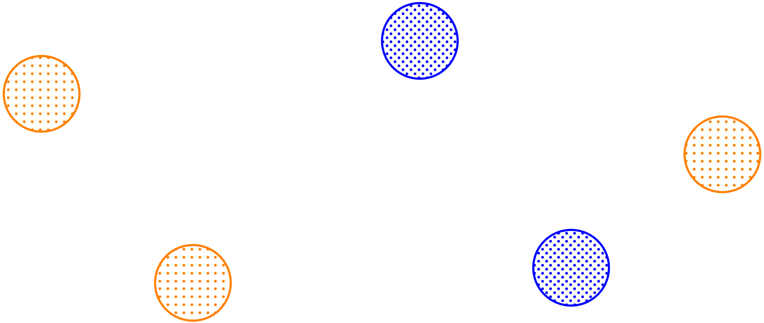


# Example: learning labels for well-separated clusters

**SETTING**

Suppose that the instance space $\mathcal{X}$ consists of countably many well-separated clusters

- within-cluster distances $\ll$ between-cluster distances

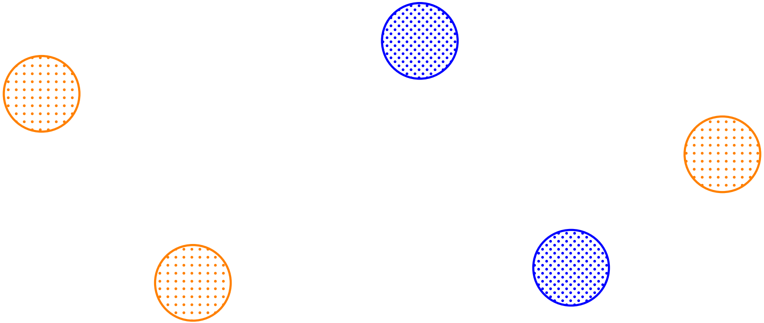


# Example: learning labels for well-separated clusters

**SETTING**

Suppose that the instance space $\mathcal{X}$ consists of countably many well-separated clusters

- within-cluster distances $\ll$ between-cluster distances
- the labels for each cluster is pure (all positive or all negative labels).



# Example: learning labels for well-separated clusters

**SETTING**

Suppose that the instance space $\mathcal{X}$ consists of countably many well-separated clusters

- within-cluster distances $\ll$ between-cluster distances
- the labels for each cluster is pure (all positive or all negative labels).

**THE NEAREST NEIGHBOR LEARNER**

makes at most one mistake made per cluster

# Example: learning labels for well-separated clusters

**CONVERGENCE RESULT FOR WELL-SEPARATED CLUSTERS**

Let $\nu$ be a finite measure on $\mathcal{X}$.

# Example: learning labels for well-separated clusters

**CONVERGENCE RESULT FOR WELL-SEPARATED CLUSTERS**

Let $\nu$ be a finite measure on $\mathcal{X}$.The nearest neighbor learner achieves vanishing error rate against any $\nu$-dominated adversary.

# Example: learning labels for well-separated clusters

Proof sketch.

- Split $\mathcal{X}$ into two pieces $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$, where:

# Example: learning labels for well-separated clusters

Proof sketch.

- Split $\mathcal{X}$ into two pieces $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$, where:
  - $\mathcal{X}_{\text{easy}}$ is the union of a finite collection of clusters

# Example: learning labels for well-separated clusters

Proof sketch.

- Split $\mathcal{X}$ into two pieces $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$, where:
  - $\mathcal{X}_{\text{easy}}$ is the union of a finite collection of clusters
  - $\mathcal{X}_{\text{small}}$ satisfies $\nu(\mathcal{X}_{\text{small}}) < \delta$.

# Example: learning labels for well-separated clusters

Proof sketch.

- Split $\mathcal{X}$ into two pieces $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$, where:
  - $\mathcal{X}_{\text{easy}}$ is the union of a finite collection of clusters
  - $\mathcal{X}_{\text{small}}$ satisfies $\nu(\mathcal{X}_{\text{small}}) < \delta$.

- Nearest neighbor can only make finitely many mistakes on $\mathcal{X}_{\text{easy}}$.

# Example: learning labels for well-separated clusters

## Proof sketch.

- Split $\mathcal{X}$ into two pieces $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$, where:
  - $\mathcal{X}_{\text{easy}}$ is the union of a finite collection of clusters
  - $\mathcal{X}_{\text{small}}$ satisfies $\nu(\mathcal{X}_{\text{small}}) < \delta$.
- Nearest neighbor can only make finitely many mistakes on $\mathcal{X}_{\text{easy}}$.
  - These mistakes contribute nothing to the asymptotic mistake rate.

# Example: learning labels for well-separated clusters

## Proof sketch.

- Split $\mathcal{X}$ into two pieces $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$, where:
  - $\mathcal{X}_{\text{easy}}$ is the union of a finite collection of clusters
  - $\mathcal{X}_{\text{small}}$ satisfies $\nu(\mathcal{X}_{\text{small}}) < \delta$.
- Nearest neighbor can only make finitely many mistakes on $\mathcal{X}_{\text{easy}}$.
  - These mistakes contribute nothing to the asymptotic mistake rate.
- The $\nu$-dominated adversary selects points from $\mathcal{X}_{\text{small}}$ at rate $\mu(\mathcal{X}_{\text{small}}) < \varepsilon$.

# Example: learning labels for well-separated clusters

## Proof sketch.

- Split $\mathcal{X}$ into two pieces $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$, where:
  - $\mathcal{X}_{\text{easy}}$ is the union of a finite collection of clusters
  - $\mathcal{X}_{\text{small}}$ satisfies $\nu(\mathcal{X}_{\text{small}}) < \delta$.
- Nearest neighbor can only make finitely many mistakes on $\mathcal{X}_{\text{easy}}$.
  - These mistakes contribute nothing to the asymptotic mistake rate.
- The $\nu$-dominated adversary selects points from $\mathcal{X}_{\text{small}}$ at rate $\mu(\mathcal{X}_{\text{small}}) < \varepsilon$.
  - By the law of large number, at most an $\varepsilon$-fraction of $(x_t)_t$ comes from $\mathcal{X}_{\text{small}}$:

$$\limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{1}\{x_t \in \mathcal{X}_{\text{small}}\} = \limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mu_t(\mathcal{X}_{\text{small}}) < \varepsilon \quad \text{a.s.}$$

# Example: learning labels for well-separated clusters

## Proof sketch.

- Split $\mathcal{X}$ into two pieces $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$, where:
  - $\mathcal{X}_{\text{easy}}$ is the union of a finite collection of clusters
  - $\mathcal{X}_{\text{small}}$ satisfies $\nu(\mathcal{X}_{\text{small}}) < \delta$.

- Nearest neighbor can only make finitely many mistakes on $\mathcal{X}_{\text{easy}}$.
  - These mistakes contribute nothing to the asymptotic mistake rate.

- The $\nu$-dominated adversary selects points from $\mathcal{X}_{\text{small}}$ at rate $\mu(\mathcal{X}_{\text{small}}) < \varepsilon$.
  - By the law of large number, at most an $\varepsilon$-fraction of $(x_t)_t$ comes from $\mathcal{X}_{\text{small}}$:

$$\limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{1}\{x_t \in \mathcal{X}_{\text{small}}\} = \limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mu_t(\mathcal{X}_{\text{small}}) < \varepsilon \quad \text{a.s.}$$

Thus, the asymptotic mistake rate is upper bounded by any $\varepsilon > 0$ by selecting $\mathcal{X}_{\text{small}}$ sufficiently small.

# Example: learning labels for well-separated clusters

## Proof sketch.

- Split $\mathcal{X}$ into two pieces $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$, where:
  - $\mathcal{X}_{\text{easy}}$ is the union of a finite collection of clusters
  - $\mathcal{X}_{\text{small}}$ satisfies $\nu(\mathcal{X}_{\text{small}}) < \delta$.
- Nearest neighbor can only make finitely many mistakes on $\mathcal{X}_{\text{easy}}$.
  - These mistakes contribute nothing to the asymptotic mistake rate.
- The $\nu$-dominated adversary selects points from $\mathcal{X}_{\text{small}}$ at rate $\mu(\mathcal{X}_{\text{small}}) < \varepsilon$.
  - By the law of large number, at most an $\varepsilon$-fraction of $(x_t)_t$ comes from $\mathcal{X}_{\text{small}}$:

$$\limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{1}\{x_t \in \mathcal{X}_{\text{small}}\} = \limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mu_t(\mathcal{X}_{\text{small}}) < \varepsilon \quad \text{a.s.}$$

Thus, the asymptotic mistake rate is upper bounded by any $\varepsilon > 0$ by selecting $\mathcal{X}_{\text{small}}$ sufficiently small. The asymptotic mistake rate is zero by taking $\varepsilon \downarrow 0$. $\square$

# Generalizing the argument

The argument works even if the clusters are not well-separated.

# Generalizing the argument

The argument works even if the clusters are not well-separated.

KEY PROPERTY USED

The nearest neighbor learner makes at most one mistake per cluster.

# Generalizing the argument

The argument works even if the clusters are not well-separated.

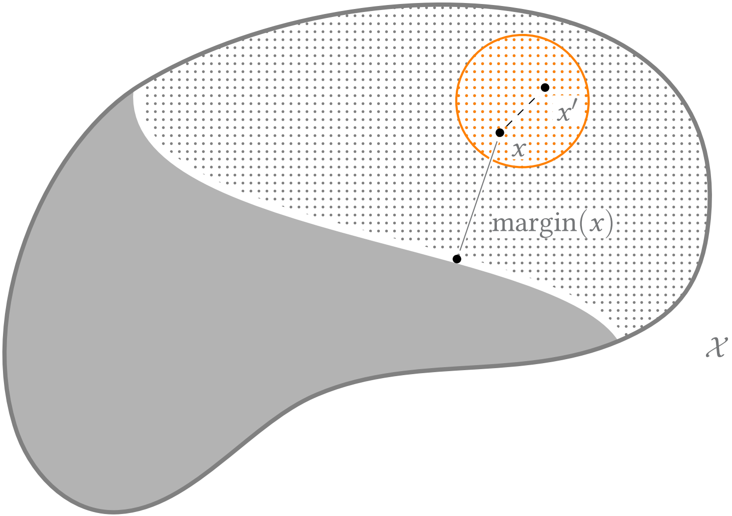
KEY PROPERTY USED

The nearest neighbor learner makes at most one mistake per mutually-labeling set.

- We introduce the device of mutually-labeling sets $U \subset \mathcal{X}$ satisfying the property:

$$\text{interpoint distances in } U < \text{distance to points with different labels.}$$

# Mutually-labeling set



# Generalizing argument

## Definition (Mutually-labeling set)

*A subset $U \subset \mathcal{X}$ is* ***mutually labeling*** *if for all $x, x' \in U$:*

$$\underbrace{\rho(x, x')}_{\textit{interpoint distances}} \quad < \quad \underbrace{\mathrm{margin}(x)}_{\textit{distance to decision boundary}}$$

*where* $\mathrm{margin}(x)$ *is the smallest distance between $x$ and points with different labels:*

$$\mathrm{margin}(x) = \inf\{\rho(x, \overline{x}) : f(x) \neq f(\overline{x})\}.$$

# Convergence result

**SETTING**

Let $(\mathcal{X}, \rho, \nu)$ be a space equipped with a separable metric $\rho$ and a finite Borel measure $\nu$.

# Convergence result

**SETTING**

Let $(\mathcal{X}, \rho, \nu)$ be a space equipped with a separable metric $\rho$ and a finite Borel measure $\nu$.

- Let $f : \mathcal{X} \to \mathcal{Y}$ have a boundary set $\partial\mathcal{X}$ with $\nu$-measure zero, where:

$$\partial\mathcal{X} := \{\mathrm{margin}(x) = 0\}.$$

# Convergence result

**SETTING**

Let $(\mathcal{X}, \rho, \nu)$ be a space equipped with a separable metric $\rho$ and a finite Borel measure $\nu$.

- Let $f : \mathcal{X} \to \mathcal{Y}$ have a boundary set $\partial\mathcal{X}$ with $\nu$-measure zero, where:

$$\partial\mathcal{X} := \{\mathrm{margin}(x) = 0\}.$$

## Theorem (Convergence of nearest neighbor)

*The nearest neighbor rule achieves vanishing mistake rate against a $\nu$-dominated adversary:*

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}\{\hat{y}_t \neq y_t\} = 0 \quad \text{a.s.}$$

# Convergence result

Proof sketch.

- Sufficiently small open balls around non-boundary points are mutually-labeling.

# Convergence result

Proof sketch.

- Sufficiently small open balls around non-boundary points are mutually-labeling.
- There is an a.e.-countable cover of $\mathcal{X}$ by these balls:

# Convergence result

Proof sketch.

- Sufficiently small open balls around non-boundary points are mutually-labeling.
- There is an a.e.-countable cover of $\mathcal{X}$ by these balls:
  - Use separability of $\mathcal{X}$ and that $\partial\mathcal{X}$ has measure zero.

# Convergence result

Proof sketch.

- Sufficiently small open balls around non-boundary points are mutually-labeling.
- There is an a.e.-countable cover of $\mathcal{X}$ by these balls:
  - Use separability of $\mathcal{X}$ and that $\partial\mathcal{X}$ has measure zero.
- Decompose $\mathcal{X}$ into $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$ like before:

# Convergence result

**Proof sketch.**

- Sufficiently small open balls around non-boundary points are mutually-labeling.
- There is an a.e.-countable cover of $\mathcal{X}$ by these balls:
  - Use separability of $\mathcal{X}$ and that $\partial\mathcal{X}$ has measure zero.
- Decompose $\mathcal{X}$ into $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$ like before:
  - $\mathcal{X}_{\text{easy}}$ is a finite union of these balls.

## Convergence result

### Proof sketch.

- Sufficiently small open balls around non-boundary points are mutually-labeling.
- There is an a.e.-countable cover of $\mathcal{X}$ by these balls:
  - Use separability of $\mathcal{X}$ and that $\partial\mathcal{X}$ has measure zero.
- Decompose $\mathcal{X}$ into $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$ like before:
  - $\mathcal{X}_{\text{easy}}$ is a finite union of these balls.
  - $\mathcal{X}_{\text{small}}$ contains very little mass $\nu(\mathcal{X}_{\text{small}}) < \delta$.

# Convergence result

**Proof sketch.**

- Sufficiently small open balls around non-boundary points are mutually-labeling.
- There is an a.e.-countable cover of $\mathcal{X}$ by these balls:
  - Use separability of $\mathcal{X}$ and that $\partial\mathcal{X}$ has measure zero.
- Decompose $\mathcal{X}$ into $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$ like before:
  - $\mathcal{X}_{\text{easy}}$ is a finite union of these balls.
  - $\mathcal{X}_{\text{small}}$ contains very little mass $\nu(\mathcal{X}_{\text{small}}) < \delta$.
    - Such a decomposition exists for any $\delta > 0$ by the finiteness of $\nu$.

# Convergence result

Proof sketch.

- Sufficiently small open balls around non-boundary points are mutually-labeling.
- There is an a.e.-countable cover of $\mathcal{X}$ by these balls:
  - Use separability of $\mathcal{X}$ and that $\partial\mathcal{X}$ has measure zero.
- Decompose $\mathcal{X}$ into $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$ like before:
  - $\mathcal{X}_{\text{easy}}$ is a finite union of these balls.
  - $\mathcal{X}_{\text{small}}$ contains very little mass $\nu(\mathcal{X}_{\text{small}}) < \delta$.
    - Such a decomposition exists for any $\delta > 0$ by the finiteness of $\nu$.
- The asymptotic mistake rate is upper bounded by any $\varepsilon > 0$ a.s. (prior argument).

# Convergence result

**Proof sketch.**

- Sufficiently small open balls around non-boundary points are mutually-labeling.
- There is an a.e.-countable cover of $\mathcal{X}$ by these balls:
  - Use separability of $\mathcal{X}$ and that $\partial\mathcal{X}$ has measure zero.
- Decompose $\mathcal{X}$ into $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$ like before:
  - $\mathcal{X}_{\text{easy}}$ is a finite union of these balls.
  - $\mathcal{X}_{\text{small}}$ contains very little mass $\nu(\mathcal{X}_{\text{small}}) < \delta$.
    - Such a decomposition exists for any $\delta > 0$ by the finiteness of $\nu$.
- The asymptotic mistake rate is upper bounded by any $\varepsilon > 0$ a.s. (prior argument).
  - Upper bounds for any countable collection of $\varepsilon_k \downarrow 0$ hold simultaneously.

# Convergence result

## Proof sketch.

- Sufficiently small open balls around non-boundary points are mutually-labeling.
- There is an a.e.-countable cover of $\mathcal{X}$ by these balls:
  - Use separability of $\mathcal{X}$ and that $\partial\mathcal{X}$ has measure zero.
- Decompose $\mathcal{X}$ into $\mathcal{X}_{\text{easy}} \cup \mathcal{X}_{\text{small}}$ like before:
  - $\mathcal{X}_{\text{easy}}$ is a finite union of these balls.
  - $\mathcal{X}_{\text{small}}$ contains very little mass $\nu(\mathcal{X}_{\text{small}}) < \delta$.
    - Such a decomposition exists for any $\delta > 0$ by the finiteness of $\nu$.
- The asymptotic mistake rate is upper bounded by any $\varepsilon > 0$ a.s. (prior argument).
  - Upper bounds for any countable collection of $\varepsilon_k \downarrow 0$ hold simultaneously.

Thus, the mistake rate converges to zero almost surely. $\square$

# Discussion about this work

**UPSHOT**

1. The nearest neighbor rule works fine against the $\nu$-dominated adversary.

# Discussion about this work

**UPSHOT**

1. The nearest neighbor rule works fine against the $\nu$-dominated adversary.
   - Along the way, we obtained a nice analytic tool (mutually-labeling sets).

# Discussion about this work

**UPSHOT**

1. The nearest neighbor rule works fine against the $\nu$-dominated adversary.
   - Along the way, we obtained a nice analytic tool (mutually-labeling sets).

2. The argument generalizes to other certain types of online learners (see paper).

# Discussion about this work

**UPSHOT**

1. The nearest neighbor rule works fine against the $\nu$-dominated adversary.
   - Along the way, we obtained a nice analytic tool (mutually-labeling sets).
2. The argument generalizes to other certain types of online learners (see paper).
   - We give a sufficient condition to online learning against a dominated adversary.

# Discussion about this work

**UPSHOT**

1. The nearest neighbor rule works fine against the $\nu$-dominated adversary.
   - Along the way, we obtained a nice analytic tool (mutually-labeling sets).

2. The argument generalizes to other certain types of online learners (see paper).
   - We give a sufficient condition to online learning against a dominated adversary.

**QUESTIONS**

- Does the $\nu$-dominated adversary balance between generality and tractability well?

# Discussion about this work

**UPSHOT**

1. The nearest neighbor rule works fine against the $\nu$-dominated adversary.
   - Along the way, we obtained a nice analytic tool (mutually-labeling sets).
2. The argument generalizes to other certain types of online learners (see paper).
   - We give a sufficient condition to online learning against a dominated adversary.

**QUESTIONS**

- Does the $\nu$-dominated adversary balance between generality and tractability well?
- Can the arguments still hold when there is benign label noise?

# Discussion about this work

**UPSHOT**

1. The nearest neighbor rule works fine against the $\nu$-dominated adversary.
    - Along the way, we obtained a nice analytic tool (mutually-labeling sets).
2. The argument generalizes to other certain types of online learners (see paper).
    - We give a sufficient condition to online learning against a dominated adversary.

**QUESTIONS**

- Does the $\nu$-dominated adversary balance between generality and tractability well?
- Can the arguments still hold when there is benign label noise?
- What do meaningful rates of convergence look like?

# Big picture: smoothed analysis for online learning

Many applications of machine learning happen in the online setting:

# Big picture: smoothed analysis for online learning

Many applications of machine learning happen in the online setting:

- never-ending and non-i.i.d. stream of tasks

# Big picture: smoothed analysis for online learning

Many applications of machine learning happen in the online setting:

- never-ending and non-i.i.d. stream of tasks
- models are updated incrementally

# Big picture: smoothed analysis for online learning

Many applications of machine learning happen in the online setting:

- never-ending and non-i.i.d. stream of tasks
- models are updated incrementally

OPPORTUNITY: we might not live in the worst-case adversarial setting

# Big picture: smoothed analysis for online learning

Many applications of machine learning happen in the online setting:

- never-ending and non-i.i.d. stream of tasks
- models are updated incrementally

OPPORTUNITY: we might not live in the worst-case adversarial setting

- Is the $\nu$-dominated online learning setting realistic and tractable?

# Big picture: smoothed analysis for online learning

Many applications of machine learning happen in the online setting:

- never-ending and non-i.i.d. stream of tasks
- models are updated incrementally

OPPORTUNITY: we might not live in the worst-case adversarial setting

- Is the $\nu$-dominated online learning setting realistic and tractable?
- If so, can we design and analyze algorithms specifically for this setting?
  - e.g. a minimax optimal algorithm might not be optimal in this setting

# Thank you

**ACKNOWLEDGEMENTS**

Joint work with Sanjoy Dasgupta.

This work is under review, but send me an email for paper/further discussion: *agso@eng.ucsd.edu*.

# References


Nika Haghtalab, Tim Roughgarden, and Abhishek Shetty. Smoothed analysis of online and differentially private learning. *Advances in Neural Information Processing Systems*, 33:9203–9215, 2020.

Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning: Stochastic and constrained adversaries. *arXiv preprint arXiv:1104.5070*, 2011.

Daniel A Spielman and Shang-Hua Teng. Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *Journal of the ACM (JACM)*, 51(3):385–463, 2004.