# ONLINE CONVEX OPTIMIZATION

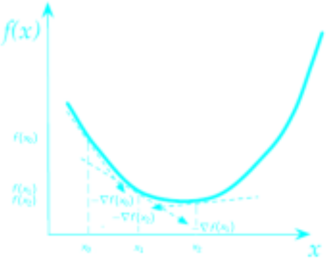
# Advanced Machine Learning

- One of the most relevant instantiations of the online learning problem is the problem of *online convex optimization* (OCO), which is characterized by a loss function

$$\ell : \mathcal{A} \times \mathcal{Z} \to \mathbb{R},$$

## Online Convex Optimization - Part 1

which is convex w.r.t. the action, i.e., $a \mapsto \ell(a, z)$ is convex for any $z \in \mathcal{Z}$.




**Learning goals**

- Get to know the class of online convex optimization problems
- Derive the online gradient descent as a suitable learning algorithm for such cases

# ONLINE CONVEX OPTIMIZATION

- One of the most relevant instantiations of the online learning problem is the problem of *online convex optimization* (OCO), which is characterized by a loss function

$$L : \mathcal{A} \times \mathcal{Z} \to \mathbb{R},$$

which is convex w.r.t. the action, i.e., $a \mapsto (a, z)$ is convex for any $z \in \mathcal{Z}$.

- Note that both OLO and OQO belong to the class of online convex optimization problems:
  - *Online linear optimization (OLO) with convex action spaces:*

$$(a, z) = a^\top z$$

  is a convex function in $a \in \mathcal{A}$, provided $\mathcal{A}$ is convex.
  - *Online quadratic optimization (OQO) with convex action spaces:*

$$(a, z) = \frac{1}{2} ||a - z||_2^2$$

  is a convex function in $a \in \mathcal{A}$, provided $\mathcal{A}$ is convex.

# ONLINE GRADIENT DESCENT: MOTIVATION

- We have seen that the FTRL algorithm with the $_2$ norm regularization $\psi(a) = \frac{1}{2\eta}\|a\|_2^2$ achieves satisfactory results for online linear optimization (OLO) problems, that is, if $(a, z) = L^{\mathtt{lin}}(a, z) := a^\top z$, then we have
  - *Fast updates* — If $\mathcal{A} = \mathbb{R}^d$, then

$$a_{t+1}^{\mathrm{FTRL}} = a_t^{\mathrm{FTRL}} - \eta z_t, \quad t = 1, \ldots, T;$$

  - *Regret bounds* — By an appropriate choice of $\eta$ and some (mild) assumptions on $\mathcal{A}$ and $\mathcal{Z}$, we have

$$R_T^{\mathrm{FTRL}} = o(T).$$

- One of the most relevant instantiations of the online learning problem is the problem of *online convex optimization* (OCO), which is characterized by a loss function

$$L : \mathcal{A} \times \mathcal{Z} \to \mathbb{R},$$

which is convex w.r.t. the action, i.e., $a \mapsto L(a, z)$ is convex for any $z \in \mathcal{Z}$.

- Note that both OLO and OQO belong to the class of online convex optimization problems:
  - *Online linear optimization (OLO) with convex action spaces:*

$$L(a, z) = a^\top z$$

  is a convex function in $a \in \mathcal{A}$, provided $\mathcal{A}$ is convex.
  - *Online quadratic optimization (OQO) with convex action spaces:*

$$L(a, z) = \frac{1}{2}\|a - z\|_2^2$$

  is a convex function in $a \in \mathcal{A}$, provided $\mathcal{A}$ is convex.

# ONLINE GRADIENT DESCENT: MOTIVATION

- We have seen that the FTRL algorithm with the $L_2$ norm regularization $\psi(a) = \frac{1}{2}\|a\|^2$ achieves satisfactory results for online linear optimization (OLO) problems, that is, if $L(a, z) = L^{\mathtt{lin}}(a, z) := a^\top z$, then we have
  - *Fast updates* — If $\mathcal{A} = \mathbb{R}^d$, then

$$a_{t+1}^{\mathrm{FTRL}} = a_t^{\mathrm{FTRL}} - \eta\, z_t, \qquad t = 1, \ldots, T;$$

  - *Regret bounds* — By an appropriate choice of $\eta$ and some (mild) assumptions on $\mathcal{A}$ and $\mathcal{Z}$, we have

$$R_T^{\mathrm{FTRL}} = o(T).$$

Apparently, the nice form of the loss function $L^{\mathtt{lin}}$ is responsible for the appealing properties of FTRL in this case. Indeed, since $\nabla_a L^{\mathtt{lin}}(a, z) = z$ note that the update rule can be written as $a_{t+1}^{\mathrm{FTRL}} = a_t^{\mathrm{FTRL}} - \eta\, z_t = a_t^{\mathrm{FTRL}} - \eta\, \nabla_a L^{\mathtt{lin}}(a_t^{\mathrm{FTRL}}, z_t)$.

# ONLINE GRADIENT DESCENT: MOTIVATION

Apparently, the nice form of the loss function $L^{\mathtt{lin}}$ is responsible for the appealing properties of FTRL in this case. Indeed, since $\nabla_a L^{\mathtt{lin}}(a, z) = z$ note that the update rule can be written as

$$a_{t+1}^{\mathrm{FTRL}} = a_t^{\mathrm{FTRL}} - \eta\, z_t = a_t^{\mathrm{FTRL}} - \eta\, \nabla_a L^{\mathtt{lin}}(a_t^{\mathrm{FTRL}}, z_t).$$

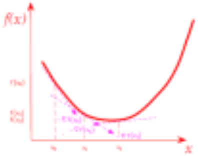
*Interpretation*: In each time step $t + 1$, we are following the direction with the steepest decrease of the most recent loss (represented by $-\nabla L^{\mathtt{lin}}(a_t^{\mathrm{FTRL}}, z_t)$) from the current "position" $a_t^{\mathrm{FTRL}}$ with the step size $\eta$

# ONLINE GRADIENT DESCENT: MOTIVATION

Apparently, the nice form of the loss function $L^{\mathtt{lin}}$ is responsible for the appealing properties of FTRL in this case. Indeed, since $\nabla_a L^{\mathtt{lin}}(a, z) = z$ note that the update rule can be written as

$$a_{t+1}^{\mathrm{FTRL}} = a_t^{\mathrm{FTRL}} - \eta\, z_t = a_t^{\mathrm{FTRL}} - \eta\, \nabla_a L^{\mathtt{lin}}(a_t^{\mathrm{FTRL}}, z_t).$$

*Interpretation*: In each time step $t + 1$, we are following the direction with the steepest decrease of the most recent loss (represented by $-\nabla L^{\mathtt{lin}}(a_t^{\mathrm{FTRL}}, z_t)$) from the current "position" $a_t^{\mathrm{FTRL}}$ with the step size $\eta$




$\Rightarrow$ Gradient Descent.

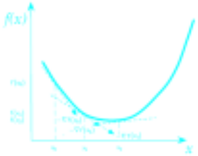
# ONLINE GRADIENT DESCENT: MOTIVATION

- **Question:** How to transfer this idea of the Gradient Descent for the update formula to other loss functions, while still preserving the regret bounds?

Apparently, the nice form of the loss function $L^{\mathtt{lin}}$ is responsible for the appealing properties of FTRL in this case. Indeed, since $\nabla_a L^{\mathtt{lin}}(a, z) = z$ note that the update rule can be written as

$$a_{t+1}^{\mathrm{FTRL}} = a_t^{\mathrm{FTRL}} - \eta\, z_t = a_t^{\mathrm{FTRL}} - \eta\, \nabla_a L^{\mathtt{lin}}(a_t^{\mathrm{FTRL}}, z_t).$$

*Interpretation*: In each time step $t+1$, we are following the direction with the steepest decrease of the most recent loss (represented by $-\nabla L^{\mathtt{lin}}(a_t^{\mathrm{FTRL}}, z_t)$) from the current "position" $a_t^{\mathrm{FTRL}}$ with the step size $\eta$




$\Rightarrow$ Gradient Descent.

# ONLINE GRADIENT DESCENT: MOTIVATION

- **Question:** How to transfer this idea of the Gradient Descent for the update formula to other loss functions, while still preserving the regret bounds?
- **Solution (for convex losses):** Recall the equivalent characterization of convexity of differentiable convex functions:

$$
\begin{aligned}
f : S \to \mathbb{R} \text{ is convex } &\Leftrightarrow f(y) \geq f(x) + (y - x)^\top \nabla f(x) \text{ for any } x, y \in S \\
&\Leftrightarrow f(x) - f(y) \leq (x - y)^\top \nabla f(x) \text{ for any } x, y \in S.
\end{aligned}
$$

# ONLINE GRADIENT DESCENT: MOTIVATION

- **Question:** How to transfer this idea of the Gradient Descent for the update formula to other loss functions, while still preserving the regret bounds?
- **Solution (for convex losses):** Recall the equivalent characterization of convexity of differentiable convex functions:

$$
\begin{aligned}
f : S \to \mathbb{R} \text{ is convex } &\Leftrightarrow f(y) \geq f(x) + (y - x)^\top \nabla f(x) \text{ for any } x, y \in S \\
&\Leftrightarrow f(x) - f(y) \leq (x - y)^\top \nabla f(x) \text{ for any } x, y \in S.
\end{aligned}
$$

- This means if we are dealing with a loss function $: \mathcal{A} \times \mathcal{Z} \to \mathbb{R}$, which is convex and differentiable in its first argument ($\mathcal{A}$ has also to be convex), then

$$
(a, z) - (\breve{a}, z) \leq (a - \breve{a})^\top \nabla_a(a, z), \quad \forall a, \breve{a} \in \mathcal{A}, z \in \mathcal{Z}.
$$

# ONLINE GRADIENT DESCENT: MOTIVATION

**Reminder:** $L(a,z) - L(\tilde{a},z) \leq (a-\tilde{a})^\top \nabla_a L(a,z), \quad \forall a,\tilde{a} \in \mathcal{A}, z \in \mathcal{Z}.$

- **Question:** How to transfer this idea of the Gradient Descent for the update formula to other loss functions, while still preserving the regret bounds?
- **Solution (for convex losses):** Recall the equivalent characterization of convexity of differentiable convex functions:

$$
\begin{aligned}
f : S \to \mathbb{R} \text{ is convex } &\Leftrightarrow f(y) \geq f(x) + (y-x)^\top \nabla f(x) \text{ for any } x, y \in S \\
&\Leftrightarrow f(x) - f(y) \leq (x-y)^\top \nabla f(x) \text{ for any } x, y \in S.
\end{aligned}
$$

- This means if we are dealing with a loss function $L : \mathcal{A} \times \mathcal{Z} \to \mathbb{R}$, which is convex and differentiable in its first argument ($\mathcal{A}$ has also to be convex), then

$$
L(a,z) - L(\tilde{a},z) \leq (a-\tilde{a})^\top \nabla_a L(a,z), \quad \forall a,\tilde{a} \in \mathcal{A}, z \in \mathcal{Z}.
$$

# ONLINE GRADIENT DESCENT: MOTIVATION

**Reminder:** $(a, z) - (\tilde{a}, z) \leq (a - \tilde{a})^{\top} \nabla_a (a, z) \; \forall a, \tilde{a} \in \mathcal{A}, z \in \mathcal{Z}.$

- Let $z_1, \ldots, z_T$ arbitrary environmental data and $a_1, \ldots, a_T$ be some arbitrary action sequence. Substitute $\tilde{z}_t := \nabla_a(a_t, z_t)$ and note that

# ONLINE GRADIENT DESCENT: MOTIVATION

**Reminder:** $(a,z) - (\tilde{a}, z) \le (a - \tilde{a})^\top \nabla_a (a,z) \; \forall a, \tilde{a} \in \mathcal{A}, z \in \mathcal{Z}.$

- Let $z_1, \ldots, z_T$ arbitrary environmental data and $a_1, \ldots, a_T$ be some arbitrary action sequence. Substitute $\tilde{z}_t := \nabla_a (a_t, z_t)$ and note that

$$R_T(\tilde{a}) = \sum_{t=1}^{T} (a_t, z_t) - (\tilde{a}, z_t) \le \sum_{t=1}^{T} (a_t - \tilde{a})^\top \nabla_a (a_t, z_t)$$

$$= \sum_{t=1}^{T} (a_t - \tilde{a})^\top \tilde{z}_t = \sum_{t=1}^{T} a_t^\top \tilde{z}_t - \tilde{a}^\top \tilde{z}_t = \sum_{t=1}^{T} L^{\mathrm{lin}}(a_t, \tilde{z}_t) - L^{\mathrm{lin}}(\tilde{a}, \tilde{z}_t).$$

# ONLINE GRADIENT DESCENT: MOTIVATION

**Reminder:** $L(a,z) - L(\tilde{a},z) \le (a-\tilde{a})^\top \nabla_a L(a,z) \ \forall a,\tilde{a} \in A, z \in Z.$

- Let $z_1, \ldots, z_T$ arbitrary environmental data and $a_1, \ldots, a_T$ be some arbitrary action sequence. Substitute $\tilde{z}_t := \nabla_a L(a_t, z_t)$ and note that

$$R_T(\tilde{a}) = \sum_{t=1}^{T} L(a_t, z_t) - L(\tilde{a}, z_t) \le \sum_{t=1}^{T} (a_t - \tilde{a})^\top \nabla_a L(a_t, z_t)$$

$$= \sum_{t=1}^{T} (a_t - \tilde{a})^\top \tilde{z}_t = \sum_{t=1}^{T} a_t^\top \tilde{z}_t - \tilde{a}^\top \tilde{z}_t = \sum_{t=1}^{T} L^{\mathrm{lin}}(a_t, \tilde{z}_t) - L^{\mathrm{lin}}(\tilde{a}, \tilde{z}_t).$$

*Conclusion:* The regret of a learner with respect to a differentiable and convex loss function is bounded by the regret corresponding to an online linear optimization problem with environmental data $\tilde{z}_t = \nabla_a L(a_t, z_t)$.

# ONLINE GRADIENT DESCENT: MOTIVATION

**Reminder:** $\ell(a, z) - \ell(\tilde{a}, z) \leq (a - \tilde{a})^\top \nabla_a \ell(a, z) \; \forall a, \tilde{a} \in \mathcal{A}, z \in \mathcal{Z}.$

- Let $z_1, \ldots, z_T$ arbitrary environmental data and $a_1, \ldots, a_T$ be some arbitrary action sequence. Substitute $\tilde{z}_t := \nabla_a \ell(a_t, z_t)$ and note that

$$R_T(\tilde{a}) = \sum_{t=1}^{T} \ell(a_t, z_t) - \ell(\tilde{a}, z_t) \leq \sum_{t=1}^{T} (a_t - \tilde{a})^\top \nabla_a \ell(a_t, z_t)$$

$$= \sum_{t=1}^{T} (a_t - \tilde{a})^\top \tilde{z}_t = \sum_{t=1}^{T} a_t^\top \tilde{z}_t - \tilde{a}^\top \tilde{z}_t = \sum_{t=1}^{T} L^{\text{lin}}(a_t, \tilde{z}_t) - L^{\text{lin}}(\tilde{a}, \tilde{z}_t).$$

*Conclusion:* The regret of a learner with respect to a differentiable and convex loss function $\ell$ is bounded by the regret corresponding to an online linear optimization problem with environmental data $\tilde{z}_t = \nabla_a \ell(a_t, z_t)$.

- **We know:** Online linear optimization problems can be tackled by means of the FTRL algorithm!

# ONLINE GRADIENT DESCENT: MOTIVATION

**Reminder:** $L(a,z) - L(\tilde{a},z) \leq (a-\tilde{a})^\top \nabla_a L(a,z) \quad \forall a,\tilde{a} \in \mathcal{A}, z \in \mathcal{Z}.$

- Let $z_1,\ldots,z_T$ arbitrary environmental data and $a_1,\ldots,a_T$ be some arbitrary action sequence. Substitute $\tilde{z}_t := \nabla_a(a_t,z_t)$ and note that

$$R_T(\tilde{a}) = \sum_{t=1}^{T} L(a_t,z_t) - L(\tilde{a},z_t) \leq \sum_{t=1}^{T} (a_t-\tilde{a})^\top \nabla_a L(a_t,z_t)$$

$$= \sum_{t=1}^{T} (a_t-\tilde{a})^\top \tilde{z}_t = \sum_{t=1}^{T} a_t^\top \tilde{z}_t - \tilde{a}^\top \tilde{z}_t = \sum_{t=1}^{T} L^{\mathrm{lin}}(a_t,\tilde{z}_t) - L^{\mathrm{lin}}(\tilde{a},\tilde{z}_t).$$

*Conclusion:* The regret of a learner with respect to a differentiable and convex loss function $L$ is bounded by the regret corresponding to an online linear optimization problem with environmental data $\tilde{z}_t = \nabla_a(a_t,z_t)$.

- **We know:** Online linear optimization problems can be tackled by means of the FTRL algorithm!

⇝ Incorporate the substitution $\tilde{z}_t = \nabla_a(a_t,z_t)$ into the update formula of FTRL with squared L2-norm regularization.

# ONLINE GRADIENT DESCENT: DEFINITION

- The corresponding algorithm which chooses its action according to these considerations is called the *Online Gradient Descent* (OGD) algorithm with step size $\eta > 0$. It holds in particular,

$$a^{\mathrm{OGD}}_{t+1} = a^{\mathrm{OGD}}_t - \eta \nabla_a (a^{\mathrm{OGD}}_t, z_t), \quad t = 1, \dots, T. \tag{1}$$

(Technical side note: For this update formula we assume that $\mathcal{A} = \mathbb{R}^d$. Moreover, the first action $a^{\mathrm{OGD}}_1$ is arbitrary.)

**Reminder:** $L(a, z) - L(\tilde{a}, z) \leq (a - \tilde{a})^\top \nabla_a L(a, z) \quad \forall a, \tilde{a} \in \mathcal{A}, z \in \mathcal{Z}$

- Let $z_1, \dots, z_T$ arbitrary environmental data and $a_1, \dots, a_T$ be some arbitrary action sequence. Substitute $\tilde{z}_t = \nabla_a L(a_t, z_t)$ and note that

$$R_T(\tilde{a}) = \sum_{t=1}^T L(a_t, z_t) - L(\tilde{a}, z_t) \leq \sum_{t=1}^T (a_t - \tilde{a})^\top \nabla_a L(a_t, z_t)$$

$$= \sum_{t=1}^T (a_t - \tilde{a})^\top \tilde{z}_t = \sum_{t=1}^T a_t^\top \tilde{z}_t - \tilde{a}^\top \tilde{z}_t = .$$

*Conclusion:* The regret of a learner with respect to a differentiable and convex loss function $L$ is bounded by

- **We know:** Online linear optimization problems can be tackled by means of the FTRL algorithm!

$\leadsto$ Incorporate the substitution $\tilde{z}_t = \nabla_a L(a_t, z_t)$ into the update formula of FTRL with squared L2-norm regularization.

# ONLINE GRADIENT DESCENT: DEFINITION

- The corresponding algorithm which chooses its action according to these considerations is called the *Online Gradient Descent* (OGD) algorithm with step size $\eta > 0$. It holds in particular,

$$a_{t+1}^{\text{OGD}} = a_t^{\text{OGD}} - \eta \nabla_a L(a_t^{\text{OGD}}, z_t), \quad t = 1, \dots T. \qquad (1)$$

(Technical side note: For this update formula we assume that $\mathcal{A} = \mathbb{R}^d$. Moreover, the first action $a_1^{\text{OGD}}$ is arbitrary. )