POLITECNICO DI MILANO
Mathematical Eng Master Degree
Dipartimento di Mate

Il Grinch Aveva Ragione

Artificial Intelligence and Robotic Laboratory
of Politecnico di Milano

Supervisor: Prof. Marcello Restelli

Co-supervisors: Francesco Trovò Ph.D.

Master thesis of:
Martino Bernasconi de Luca, π

Academic Year 2018-2019

Contents

List of Figures

Chapter 1

Introduction

Classical investment techniques for the portfolio management problem derive from the knowledge of the statistical distribution of the assets return. Then, once the statistical model has been chosen, the problem get solved by optimizing the expected value of the utility of some random variable (usually accounting for the trade-off between risk and return), that describes the value of the portfolio in some fixed time in the future. This line of thinking has been proposed and sustained by Markovitz, Samuelson, Fama ecc... , and it is now called Modern Portfolio Theory (MPT).

This approach is known to be very susceptible to the errors in the modelling of the random variable that model the asset return. In fact is known that the markets have a non stationary behaviour, which means that every statistical assumption is ephemeral and unreliable. and they are usually referred to to backward looking, i.e. that they optimize

A different approach has been originated from the fields of information theory at the Bell Labs in the 1950, from the works of Shannon, Kelly and Cover. This methods were first included in the classical portfolio theory framework, under the name of Capital Growth Theory (CGT) [Hakansson et al., 1995] and then got included in the machine learning literature under the framework of Online Game Playing. Only recently this field has been taken into the Online Optimization This formulation has very interesting properties such as stability in a game theory fashion (equilibrium) and robustness versus adversarial manipulation.

One of the strongest points in favor of this techniques are the strong theoretical guarantees that algorithms developed under this framework can give. This guarantees come from the game theory concept of Regret, which is a form of dissatisfaction originated from having taken an action, instead of another action.

Principal in this thesis will be the extension of the modelling of the financial applications of this methodologies to the presence of transaction costs and to provide strong theoretical assurance even in the presence of transaction costs. In fact in many financial situations, transaction costs are not modelled and

Chapter 2

Online Learning

Online Learning is a theoretical framework to formalize a sequential decision problem in which an agent has to take consecutive actions in an environment. Every time the agent takes an action, the environment returns a loss signal (or reward depending on the convention on the sign). This framework is similar to other sequential decision problems such as Reinforcement Learning [Sutton et al.,], with the exception that the loss function is decided by an adversary which has complete knowledge of your strategy in advance, rather that be described by a stochastic probability kernel. The purpose of this section is to present the general framework of Online Game Playing and to introduce the notation necessary for the development of the theory. We will define formally the framework of Online Learning with Expert Advice, which is one the most studied framework of Online Learning, due to its ability to include many other frameworks, such as Multi Armed Bandit or Online Convex Optimization. Then we will expose the concept of regret and present the relation of Online Learning to classical repeated games from the field of Game Theory. Finally we will introduce Online Convex Optimization as a special case of Online Learning with expert advice and it's interesting relation to theoretical statistical learning. The choice of this path, from Online Learning to Online Convex Optimization, has been done to show how general and powerful is Online Learning in its simplicity, and because Online Convex Optimization is the most suitable framework to present our contribution to Online Portfolio Selection, that will be presented in later chapters.

definire h_t come la storia

maybe add DAgger

ref chapter ...

In fact, even if we will focus on the portfolio problem, the apparently simple formulation of this framework is capable to encompass many other applications and problems, such as network routing [Belmega et al., 2018] and dark pool order allocation [Agarwal et al., 2010]. A torought disserta-

tion on Online Learning can be found in [Cesa-Bianchi and Lugosi, 2006].

2.1 Online Learning

Definition 2.1.1. *(Online Game Playing). Let $\mathcal{Y}$ be the outcome space, $\mathcal{D}$ the prediction space and $f : \mathcal{D} \times \mathcal{Y} \to \mathbb{R}$ is a loss function, an Online Game is the following sequential game played by the forecaster $\mathcal{A}$ and the environment:*

For each round $t \in 1, 2, \ldots$

1. *The learner $\mathcal{A}$ choose an element of the decision space $x_t \in \mathcal{D}$*
2. *The environment choose the element $r_t \in \mathcal{Y}$, and subsequently determines the loss functions $f(\cdot, r_t)$*
3. *The agent $\mathcal{A}$ incur in a loss $f(x_t, r_t)$.*
4. *The agent update its cumulative losses $L_t = L_{t-1} + f(x_t, r_t)$ with L_0*

In Online Learning an agent $\mathcal{A}$ has to guess the outcomes of a sequence $r_1, r_2, \ldots$ of some events that are in the outcome space $\mathcal{Y}$, at each time step he will play (some time we will also say *predict*) x_t that is an element of the prediction space $\mathcal{D}$, and the environment will choose a loss function $f(\cdot, r_t)$ by determine the outcome r_t. Some times is not important to know the exact outcome of the round and so we can identify the function $f(x, r_t)$ with $f_t(x)$. The simplest case is for $\mathcal{Y} = \mathcal{D}$ and both of finite cardinality, meaning that there are only a finite number of actions that the agent $\mathcal{A}$ can choose from. We will some time refer to the Environment defined in 2.1.1 as adversarial, since no stochastic characterization is given to the outcome sequence r_t, and the analysis of the regret is done assuming a worst case scenario. Since the adversary knows the prediction x_t, before deciding the outcome r_t, absolute minimization of the loss is an hopeless task and so we have to set an easier scope. We will also present the counterexample to why the absolute minimization of the loss is an hopeless task and present the adapt minimal framework to successful Online Learning in Adversarial Environment.

2.1.1 Regret and Experts

We said that the objective of absolute loss minimization is hopeless in an adversarial framework, as the adversary can always choose the outcome r_t that maximizes the loss $f(x, r_t)$ regardless of the decision $x \in \mathcal{D}$ taken by the learner. We shall present a simple counterexample in this setting.

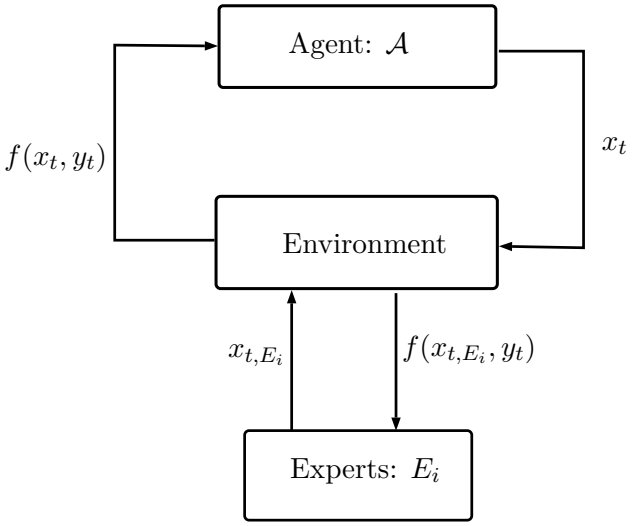

Figure 2.1: Online Learning with Expert Advice as Multi Agent-Environment interaction

Take $\mathcal{D}$ as a space of binary outcomes, *i.e.* $|\mathcal{D}| = 2$ absolute loss as $f(x, y) = |x - y|$, since the adversary plays after the learner $\mathcal{A}$, it can make the loss of the learner $L_T = T$, by outputting the bit non predicted by the learner. Notice that no assumption has been made on the strategy followed by the learner $\mathcal{A}$. From this example it is clear that the learner has to set a less ambitious goal.

We do so by extending the theoretical formulation in Section 2.1 by including a set $\mathcal{E}$ of other players, this setting is called prediction with expert advice. At each time step of the prediction game, each expert $E_i \in \mathcal{E}$, predicts an element $x_{t,i} \in \mathcal{D}$, and incurs in a loss $f(x_{t,i}, r_t)$, just as the agent $\mathcal{A}$, creating a general multi-agent interaction as in Figure 2.2. Now the goal that the learner set itself to solve is to obtain small losses with respect to the expert class $\mathcal{E}$. This concept is captured by the definition of regret. Formally we define the Regret $R_{T,i}$ for the agent $\mathcal{A}$ with respect to expert $E_i \in \mathcal{E}$ (assumed finite for the moment) as follow

$$R_{T,i} = L_T - L_{T,i} \tag{2.1}$$

The regret observed by the agent $\mathcal{A}$ with respect to the entire class of experts $\mathcal{E}$ is defined as:

$$R_T = \max_{E_i \in \mathcal{E}} R_{T,i} = L_T - \min_{E_i \in \mathcal{E}} L_{T,i} \tag{2.2}$$

The task the agent $\mathcal{A}$ is set to solve is find a sequence x_t function of the information obtained up to the time t in order to obtain small regret R_T

with respect to any sequence $r_1, r_2, \dots$ chose by the environment.

In particular we aim to achieve sub-linear regret $R_T = o(T)$, in this case the per-round regret R_T/T will asymptotically vanish

$$R_T = o(T) \implies \lim_{T\to\infty} \frac{R_T}{T} = 0, \tag{2.3}$$

where $o(T)$ is the space of sub-linear affine functions. A strategy $\mathcal{A}$ that attains sub-linear regret is call *Hannan-Consistent* [Hannan, 1957].

The regret is a measure of the distance between our online performance and the best offline (in retrospect) performance among the expert class $\mathcal{E}$, this is also called *external regret* since it is compared to the external set of experts $\mathcal{E}$. A surprising fact is even that such algorithms do even exists. In fact a first result is that in general there are no Hannan Consistent strategies, and just introducing the concept of regret is not enough for successful Online Learning:

A first simple counterexample is due to Cover [Cover, 1966], if the decision space $\mathcal{D}$ is finite then there exists a sequence of loss function such that $R_T = \Omega(T)$. Again take $\mathcal{D}$ as a space of binary outcomes, absolute loss as $f(x, y) = |x - y|$, and the class of experts is composed by two experts, one predicting always 0 and the other always 1. Taking T odd, we have that the loss of the best expert is $L_{T,i} < \frac{T}{2}$ and we already shown that the adversary can make the loss of the learner $L_T = T$. It is now evident that the regret is $R_T > T - \frac{T}{2}$, which do not allow $R_T/T \to 0$. This argument is easily extended in the case of any finite decision space $\mathcal{D}$.

In order for the learner to hope to obtain sub-linear regret is to randomize its predictions, the learner at each turn t has a probability distribution on the decision space and plays x_t according to this distribution. If the original decision space was $\mathcal{D}$ with $|\mathcal{D}| = N$ after the randomization of the decision we effectively transformed the decision space $\mathcal{D}$ into the $\Delta_{N-1} \in \mathbb{R}^N$ probability simplex, we are formally extending the game into its mixed extension, this will be discussed further in Section 2.2. It can be viewed also as a *covexification* of the domain, pointing to the undeniably necessity of convex geometry in this context, that will be discussed in 2.3. So from now on the domain $\mathcal{D}$ will be convex, either by the problem specification or by randomized convexification if the problem has discrete decision space.

2.1.2 Existence of No-Regret Startegies

The general idea with a finite class of experts is given by the Weighted Average Forecaster. A natural class of algorithm to explore is the predicting

as the weighted average prediction of the experts, weighted on the accrued regret suffered by the agent $\mathcal{A}$:

Definition 2.1.2. *(Weighted Avergae Forcester). For a finite class of experts $\mathcal{E}$ the weighted average prediction is defined as*

$$x_t = \frac{\sum_{i=1}^{N} w_{t-1,i} x_{t,i}}{\sum_{i=1}^{N} w_{t-1,i}} \tag{2.4}$$

where $w_{t-1,i} > 0$ and $x_{t,i}$ is the prediction of expert $E_i \in \mathcal{E}$ at round t.

Since $\mathcal{D}$ is convex we have that $x_t \in \mathcal{D}$. Then it's natural to assume that the weights are a function of the cumulated regret suffered by the agent, and also that the change in weight is proportional to the change in a potential function: We can generalize the simple weighted average prediction (2.1.2) in the following general form, itroduced in [Cesa-Bianchi and Lugosi, 2003]:

$$x_t = \frac{\sum_{i=1}^{N} \partial_i \mathbf{\Phi}(\mathbf{R}_{t-1}) x_{t,i}}{\sum_{i=1}^{N} \partial_i \mathbf{\Phi}(\mathbf{R}_{t-1})}, \tag{2.5}$$

where $\Phi(\mathbf{u}) = \varphi\left(\sum_{i=1}^{N} \phi(u_i)\right)$ is a function $\mathbf{\Phi} : \mathbb{R}^N \to \mathbb{R}^+$ define trhought two increasing functions $\phi, \varphi : \mathbb{R} \to \mathbb{R}^+$, $\varphi, \phi \in \mathcal{C}^2(\mathbb{R})$ and concave and convex resp. By specializing the two functions φ, ϕ we can derive most of the algorithm for dealing with prediction under expert advice. The reasons behind the general form of Equation (2.5) and an extended discussion can be found in [Hart and Mas-Colell, 2001] and [Blackwell et al., 1956], but the general idea is that the form of Equation (2.5) has the following property:

Theorem 2.1.1. *If x_t is given by Equation* (2.5) *and the loss $f(\cdot, y)$ is convex in the first argument then the instantaneous weighted regret* $\sup_{r_t \in \mathcal{Y}} \sum_{i=1}^{N} [f(x_t, r_t) - f(x_{t,i}, r_t)] \partial_i \mathbf{\Phi}(\mathbf{R}_{t-1}) \leq 0$.

Proof. By convexity of $f(\cdot, r_t)$ we have that

$$f(x_t, r_t) \leq \frac{\sum_{i=1}^{N} \partial_i \mathbf{\Phi}(\mathbf{R}_{t-1}) f(x_{t,i}, r_t)}{\sum_{i=1}^{N} \partial_i \Phi(\mathbf{R}_{t-1})}, \forall r_t \in \mathcal{Y} \tag{2.6}$$

And since $\mathbf{\Phi}(\mathbf{x}) = \varphi\left(\sum_{i=1}^{N}\phi(x_i)\right)$ we have that

$$\partial_i \mathbf{\Phi}(\mathbf{x}) = \varphi'\left(\sum_{i=1}^{N}\phi(x_i)\right)\phi'(x_i) \geq 0$$

Hence we can rearrange the terms in Equation (2.6) to obtain the statement. □

Also note that fixing the structure for the weights as in Equation (2.5) we have that $w_{t,i} \propto \phi'(R_{t,i})$ that is an increasing function in $R_{t,i}$ (since ϕ is convex and increasing) that essentially states that we are increasing the probability of playing actions on which we saw high regret $R_{t,i}$.

In the case the loss $f(\cdot, y)$ is ν-exp concave (that is $e^{-\nu f(\cdot,y)}$ is concave), then we have the following result:

Theorem 2.1.2.

$$\mathbf{\Phi}(R_T) \leq \mathbf{\Phi}(R_0)$$

where $\mathbf{\Phi}(x) = \varphi\left(\sum_{i=1}^{N}\phi(x_i)\right)$ *is chosen as* $\varphi(x) = \frac{1}{\nu}log(x)$ *and* $\phi(x) = e^{\nu x}$

Proof. The weights are given by $w_{t-1,i} = e^{\nu R_{t-1,i}} / \sum_{j=1}^{N} e^{\nu R_{t-1,j}}$. By exp-concavity we have that

$$e^{-\nu f(x_t,r_t)} = exp\left\{-\nu f\left(\frac{\sum_{i=1}^{N} w_{t-1,i}x_{t,i}}{\sum_{i=1}^{N} w_{t-1,i}}, r_t\right)\right\} \geq \frac{\sum_{i=1}^{N} w_{t-1,i}e^{-\nu f(x_{t,i},r_t)}}{\sum_{i=1}^{N} w_{t-1,i}} \quad (2.7)$$

this can be rewritten as

$$\sum_{i=1}^{N} e^{\nu R_{t-1,i}} e^{\nu[f(x_t,r_t)-f(x_{t,i},r_t)]} \leq \sum_{i=1}^{N} e^{\nu R_{t-1,i}} \quad (2.8)$$

Applying $\varphi(x) = \frac{1}{\nu}log(x)$ to both sides of equation (2.8) we obtain that

$$\Phi(\mathbf{R}_t) \leq \Phi(\mathbf{R}_{t-1})$$

that prove the thesis. □

Definition 2.1.3. *The exponentially weighted algorithm is* (2.5) *with* $\varphi(x) = \frac{1}{\eta}ln(x)$ *and* $\phi(x) = e^{\eta x}$ *giving weights of the form* $w_{t-1,i} = e^{\eta R_{t-1,i}} / \sum_{j=1}^{N} e^{\eta R_{t-1,j}}$

e allora show

It can be shown that the algorithm defined by update rule (2.1.3), and for a convex loss function $l(\cdot, y_y)$, gives the following guarantee on the regret:

$$R_T \leq \frac{log(N)}{\eta} + \frac{T\eta}{8}$$

By choosing $\eta = O\left(\sqrt{\frac{1}{T}}\right)$ we obtain a sub-linear regret $R_T = \mathcal{O}(\sqrt{T})$. It is also possible to make this algorithm an all-time algorithm (no need to know the length of the game T, as opposed as a *one-time* algorithm) by using the so called doubling trick by continually adapting the parameter η. In general a one-time algorithm obtains slightly smaller bounds then the all-time counterparts, that require the knowledge

2.1.3 Experts

The theoretical framework described in Section 2.1 is very general and most suited for a game theory analysis of the problem. This help us describe many other frameworks, such as Online Optimization, or Multi Armed Bandit (MAB) as embedded into a Game Playing framework with expert advice. It can then be specialized by fixing many elements of the definition, in order to be applied to the specific problem we are set to solve. For example the class of experts $\mathcal{E}$ is most of the time completely fictitious, meaning that the experts are not real players of the game, but most of the time they are *simulable* meaning that the agent $\mathcal{A}$ is able to compute x_{t,E_i} for each expert $E_i \in \mathcal{E}$ and most of the times the class of expert is very limited in its actions, *e.g.* $\mathcal{E}$ is the class of experts for which x_{t,E_i} is constant in t. In this case, which is the most studied class of experts, we are basically just comparing our learner $\mathcal{A}$ to the best fixed action x_t^* in hindsight. This is a clairvoyant strategy that attains the minimum cumulative loss over the entire length of the game T.

2.1.4 Uncountable Experts

In the case of uncountable experts the Exponentially Averaged Prediction cannot be applied directly, but can be extended to a continuous mixture of experts predictions. More specifically we need the case of the class $\mathcal{E}$ being generated by a convex hull of a finite number of a base class of experts. With continuous class of experts $\mathcal{E}$, defined in this way, the regret is defined as

$$R_T = L_T - \inf_{e \in \Delta_{N-1}} L_{T,q}, \tag{2.9}$$

where $\Delta_{N-1} \subset \mathbb{R}^N$ is the N-simplex, and

$$L_{T,q} = \sum_{t=1}^{T} f(\langle q, x_{t,e} \rangle, r_t)$$

and $x_{t,e} \in \mathbb{R}^N$ is the vector of expert's predictions at time t.

In this setting it seems natural to extend the Exponential Weighted Majority algorithm described by equation (2.1.2) into its continuous case:

$$x_t = \frac{\int_{\Delta_{N-1}} w_{q,t-1} < q, f > dq}{\int_{\Delta_{N-1}} w_{q,t-1} dq}$$

that can be extended to the case of potentials average as

2.2 Regret Minimization in Games

In this section we explore the connection of the framework of Section 2.1 into a more classical repeated game framework. For the moment we thought at the adversary as a black box without any specific model in mind. The reason of this chapter is to clarify its role as a player in the game and to show the game theoretical properties of no-regret agents . Since in Online Learning the convention is to speak about losses, we shall speak about losses (minimization) also in the classical definitions of game theory instead of payoffs (maximization).

no rgetet che?

Definition 2.2.1. *(Strategic Form K-Player Game). A Strategic form K-player game is t a triple $\langle \mathcal{K}, \{X_i\}_{i \in \mathcal{K}}, \{l_i\}_{i \in \mathcal{K}} \rangle$ where*

1. *$\mathcal{K} = \{1, \dots, K\}$ is the finite set of players*
2. *X_i is the set of actions available to player $i \in \mathcal{K}$*
3. *$l_i : \bigotimes_{k=1}^{K} X_i \to \mathbb{R}$ is the loss observed by player $i \in \mathcal{K}$*

The game is called finite is $|X_i| < +\infty$ for all $i \in \mathcal{K}$.

2.2.1 Mixed extension

regret?

As in Section 2.1 we saw that it is impossible to obtains good results in adversarial environment with finite action spaces. A first step to solve this has been the *randomized convexification*, where finite action spaces are extended into convex sets, given by their probability simplex, also losses are

to be interpreted as expected losses when the mixed extension is applied to the formal game. More formally:

Definition 2.2.2. *(Mixed-extension for finite games). A finite game* $\langle \mathcal{K}, \{X_i\}_{i\in\mathcal{K}}, \{l_i\}_{i\in\mathcal{K}}\rangle$ *can be extended into the game* $\langle \mathcal{K}, \{\tilde{X}_i\}_{i\in\mathcal{K}}, \{\tilde{l}_i\}_{i\in\mathcal{K}}\rangle$

1. $\tilde{X}_i = \Delta_{|X_i|-1} \subset \mathbb{R}^{|X_i|}$ *for all* $i \in \mathcal{K}$
2. $\tilde{l} : \bigotimes \tilde{X}_i \to \mathbb{R}$ *is defined as*

$$\tilde{l}(x_1, \dots, x_K) = \sum_{i_1=1}^{N} \cdots \sum_{i_K=1}^{N} p_{i_1} \dots p_{i_K} l(i_1, \dots, i_K)$$

Due to the impossibility result of Cover we have to work with the mixed extension formulation of the game. So from now on we take this step implicitly. The taxonomy of game definition is quite extended and complex, we will focus on non-cooperative games . More specifically we will need the model for *Zero Sum Game*.

non l'ho mai chiamato così

cita qualcuno

Definition 2.2.3. *(2-Player Zero-Sum Game). A Zero Sum game is a tuple* $\langle \{X_1, X_2\}, l : X_1 \times X_2 \to \mathbb{R}\rangle$. *As in Definition 2.2.1* X_1, X_2 *are the action spaces for Player* 1 *(row player) and Player 2 (columns player) respectively and* $l(x_1, x_2)$ *for* $x_i, x_2 \in X_1 \times X_2$*, represents the losses for Player* 1 *and profits for player* 2.

If this game is played for T turns, we can call it a repeated game, and the losses for each player will be $L_1^{(T)} = \sum_{t=1}^{T} l_i\left(x_i^{(t)}, x_2^{(t)}\right)$ and $L_2^{(T)} = -L_1^{(T)}$.

2.2.2 MinMax Consistency

For such games we can define a *values* for the game as:

$$V_1 = \inf_{x_1\in X_1} \sup_{x_2\in X_2} l(x_1, x_2) \tag{2.10}$$

$$V_2 = \sup_{x_2\in X_2} \inf_{x_1\in X_1} l(x_1, x_2) \tag{2.11}$$

This is essentially the Von Neumann minmax theorem, for two player zero sum game, since the probability simplex over finite action space are convex sets (in fact one could prove the minmax theorem by the existence of a Hannan consistent strategy)

These is the value that the players can guarantees themselves, meaning that no matter the strategy of the columns player, the row player could

guarantee himself a loss of at maximum V_1, the converse holds for the row player. It can be interpreted as the minimum loss (best payoff) that player could achieve if we know that the other player would play adversarially. It is clear that $V_2 \leq V_1$.

Now we will embed the framework of Online Game Playing of Section 2.1 in a two player zero sum game. Online Learning is a special form of Zero Sum Game (possibly considering its mixed extension described in 2.2.1) where $X_1 \equiv \mathcal{D}$ and $X_2 \equiv \mathcal{Y}$. The loss function $l : X_1 \times X_2 \to \mathbb{R}$ can be identified by the loss $f : \mathcal{D} \times \mathcal{Y} \to \mathbb{R}$ of the Online Learning Agent $\mathcal{A}$. Now we will explore interesting properties of Hannan Consistent strategies. A surprising fact is that if the row player plays accordingly to a Hannan Consistent strategy then it achieve the value of the game V_1. (This can be done also by playing any minmax strategy, as predicted by the Von Neumann.)

Theorem 2.2.1. *Hannan Consistent agents in Online Game Playing reach asymptotically the minmax value of the one shot game.*

$$\limsup_{T\to+\infty} \frac{1}{T} \sum_{t=1}^{T} f(x_t, y_t) \leq V_1$$

Proof. Let's suppose that player 1 plays an Hannan Consisten strategy and that $y_1, y_2, \ldots \subset \mathcal{Y}$ is a generic sequence played by the columns player.

$$\limsup_{T\to+\infty} \frac{R_T}{T} \leq 0 \tag{2.12}$$

that can be translate into

$$\limsup_{T\to+\infty} \frac{1}{T} \sum_{t=1}^{T} f(x_t, y_t) \leq \limsup_{T\to+\infty} \frac{1}{T} \min_{x\in\mathcal{D}} \sum_{t=1}^{T} f(x, y_t) \tag{2.13}$$

Lets call $\hat{y}_T$ the empirical distribution played by player 2 up to T:

$$\hat{y}_T(j) = \frac{1}{T} \sum_{t=1}^{T} y_t$$

by (2.13) we just need to show $\frac{1}{T} \min_{x\in\mathcal{D}} \sum_{t=1}^{T} f(x, y_t) \leq V$

$$\min_{x\in\mathcal{D}} \frac{1}{T} \sum_{t=1}^{T} f(x, y_t) = \min_{x\in\mathcal{D}} f(x, y_T) \leq \max_{y\in\mathcal{Y}} \min_{x\in\mathcal{D}} f(x, y) \leq V_1 \tag{2.14}$$

□

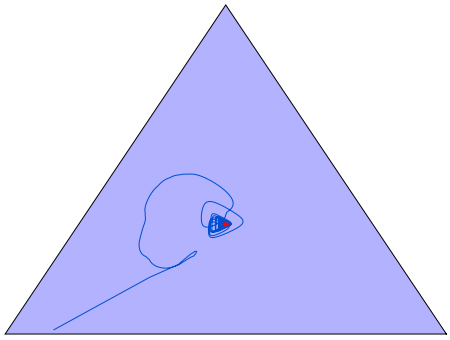

Figure 2.2: Rock Paper Scissor Dynamics Exponentially Weighted Majority

We showed that regardless of the strategy of player 2, a player playing a Hannan Consistent strategy achieves lower losses that the value of the game V_1. Clearly using a Hannan consistent strategy means that if player 2 were not adversarial, then player 1 could potentially earn a significantly higher average payoff than the value V of the game.

2.3 Online Convex Optimization for Regret Minimization

Let's compare this framework to an apparently unrelated problem, namely optimization. In online optimization an agent $\mathcal{A}$ is set to optimize a sequence of functions $f_t(x)$ where usually $f_t : \mathcal{X} \to \mathbb{R}$ is a real valued function from the set $\mathcal{X} \subset \mathbb{R}^n$. The functions $f_t(x)$ could have a persistent structure *i.e.* $f_t(x) = f(x, r_t), \forall x \in \mathcal{X}, r_t \in \mathcal{D}$, but not necessarily. The decision space $\mathcal{D}$ is assumed to be convex, as the functions $f_t : \mathcal{D}$. This

framework was first devised in [Zinkevich, 2003], and has been later wildly used in the machine learning community to engineer optimization procedures [Shalev-Shwartz et al., 2012].

Convexity plays an central role in most of the analysis made in Online Learning, and Online Convex Optimization. Convexity of the domain $\mathcal{D}$ and of the loss functions, $f(\cdot, r)$ bound the problem geometry and let us derive simple and efficient learning procedures. More generally in the subsequent section we will present the general learning

2.3.1 Statistical Learning and Online Learning

cita che fa figo

Now we explore the connection between the Online Learning framework and classical concepts of classical Statistical Learning techniques.More concretely we can prove and design a whole class of algorithm that are Agnostically PAC Learnable with Online Learning Techniques. Classical statistical learning theory deals with examples (or observations) and model of the phenomena and then uses the model to predict the future observations [Bousquet et al., 2003]. Quite informally on could say that we are trying to infer concept from examples, which is a mapping $\mathcal{C} : \mathcal{X} \to \mathcal{Y}$, where X is the domain space and $\mathcal{Y}$ is the set of labels for the examples. We then have a sample from an unknown distribution $\mathcal{D}$ such that $(x, y) \sim \mathcal{D}$, then we need to learn a mapping $h : \mathcal{X} \to \mathcal{Y}$ such that the error under the distribution $\mathcal{D}$ is small. The loss function needed to define this error is not specific to the problem and can be decided by the user, this is called generalization error and its defined as:

$$e(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[l(h(x), y)] \tag{2.15}$$

for a loss function $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$. The goal for an algorithm $\mathcal{A}$ is to produce a hypothesis h with small generalization error (2.15). It is generally difficult to generalize well and how difficult is precised by the following theorem called the *No free lunch theorem*, which states that for any learner $\mathcal{A}$ that learns an hypothesis h there exists a concept $\mathcal{C}$ with generalization error 0 and a distribution $\mathcal{D}$ such that the generalization error of $\mathcal{A}$ is at least $1/2 - \epsilon$ for any $\epsilon > 0$. Hence it is impossible to learn any concept in this sense. But we can learn concept restricting the class of concepts in a hypothesis space $\mathcal{H} : \mathcal{X} \to \mathcal{Y}$. This restriction gives raise to the concept of Probably Approximately Correct (PAC) learnability.

Definition 2.3.1. *(PAC learnable). An hypothesis class $\mathcal{H}$ is PAC learnable w.r.t. the loss l if there exists a learner $\mathcal{A}$ that seen a sample S_N of examples*

learns an hypothesis $h \in \mathcal{H}$ s.t. for all ϵ, δ there exists $N_{\epsilon,\delta}$ such that for any distribution $\mathcal{D}$ we have a generalization error $e(h) < \epsilon$ with probability $1 - \delta$

Usually we also require that the algorithm $\mathcal{A}$ learns the concept h in polynomial time w.r.t. the parameter of the problem. PAC learnability is essentially requiring that the there exists an hypothesis $h \in \mathcal{H}$ with near zero generalization error, otherwise the class $\mathcal{H}$ is not PAC learnable., otherwise the class $\mathcal{H}$ is not PAC learnable. But we can weaken the concept of PAC learnability by addressing directly this concept.

Definition 2.3.2. *(PAC agnostic learnable). Given the same definitions of definition 2.3.1, an hypothesis class $\mathcal{H}$ is PAC agnostic learnable if we have a generalization error $e(h) < \min_{\tilde{h} \in \mathcal{H}} e(\tilde{h}) + \epsilon$ with probability $1 - \delta$*

Which hypothesis spaces $\mathcal{H}$ are PAC learnable (agnostically or not) is an open and complex issue, but the case for convex hypotheses class $\mathcal{H} \in \mathcal{R}$ can be solved by Online Learning techniques, showing the versatility of the methods. Moreover the proof of such theorem gives an algorithmic perspective on the matter.

Chapter 3

Information, Prediction and Investing

In Chapter 2 we described at a high level the framework of Online Learning in Adversarial environment. Now we draw the connections between that and predictions. It surly seems counter intuitive to speak about predictions in an adversarial framework, since we are used to think about predictions only of stochastic processes. The root of this formulation are to be traced back to the Bell Laboratories in the '50, from works of Kelly [Kelly Jr, 2011], linking sequential betting and information rate. This connection is of primary importance to understand sequential investing as an instance of sequential decision problem. We first draw the parallelism between probability assignment over discrete events and Online Learning and then extend the discussion to sequential investments.

3.1 Probability assignment

The decision space $\mathcal{D}$ in the case of finite N possible bets is the $\Delta_{N-1} \subset \mathbb{R}^N$ probability simplex while the outcome $\mathcal{Y}$ space is the set $\{1, \ldots, N\}$, representing the winning bet at each turn. The loss function $f(x, y)$ should have these natural properties: low when x_y 1 and high when x_y 0 where x_y is the probability assigned to the outcome y. The inverse log-likelihood seems a reasonable proposal, simply because the multiplicative additive property of the logarithm but has also a deeper connection to information that we will discuss later on:

Definition 3.1.1. *(Self Information Loss). In the sequential probability assignment problem the loss function $f(x, y)$, $x \in \Delta_{N-1}$ and $y \in [1, \ldots, N]$*

is defined as

$$f(x, y) = -\log(x_y)$$

where x_y is the probability assigned to outcome $y \in \mathcal{Y}$.

In the case of simulable experts, the prediction x_t of the agent is a function of the history of outcomes $r^{t-1} := \{r_1, r_2, \ldots, r_{t-1}\} \in \mathcal{Y}^{t-1}$. An expert can be thought of as a set of functions $g_k : \mathcal{Y}^{k-1} \to \Delta_{N-1}$.

Bibliography

[Agarwal et al., 2010] Agarwal, A., Bartlett, P., and Dama, M. (2010). Optimal allocation strategies for the dark pool problem. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 9–16.

[Belmega et al., 2018] Belmega, E. V., Mertikopoulos, P., Negrel, R., and Sanguinetti, L. (2018). Online convex optimization and no-regret learning: Algorithms, guarantees and applications. *arXiv preprint arXiv:1804.04529*.

[Blackwell et al., 1956] Blackwell, D. et al. (1956). An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6(1):1–8.

[Bousquet et al., 2003] Bousquet, O., Boucheron, S., and Lugosi, G. (2003). Introduction to statistical learning theory. In *Summer School on Machine Learning*, pages 169–207. Springer.

[Cesa-Bianchi and Lugosi, 2003] Cesa-Bianchi, N. and Lugosi, G. (2003). Potential-based algorithms in on-line prediction and game theory. *Machine Learning*, 51(3):239–261.

[Cesa-Bianchi and Lugosi, 2006] Cesa-Bianchi, N. and Lugosi, G. (2006). *Prediction, learning, and games*. Cambridge university press.

[Cover, 1966] Cover, T. M. (1966). Behavior of sequential predictors of binary sequences. Technical report, STANFORD UNIV CALIF STANFORD ELECTRONICS LABS.

[Hakansson et al., 1995] Hakansson, N. H., Ziemba, W. T., et al. (1995). Capital growth theory. *Handbooks in operations research and management science*, 9:65–86.

[Hannan, 1957] Hannan, J. (1957). Approximation to bayes risk in repeated play. *Contributions to the Theory of Games*, 3:97–139.

[Hart and Mas-Colell, 2001] Hart, S. and Mas-Colell, A. (2001). A general class of adaptive strategies. *Journal of Economic Theory*, 98(1):26–54.

[Kelly Jr, 2011] Kelly Jr, J. L. (2011). A new interpretation of information rate. In *The Kelly Capital Growth Investment Criterion: Theory and Practice*, pages 25–34. World Scientific.

[Shalev-Shwartz et al., 2012] Shalev-Shwartz, S. et al. (2012). Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194.

[Sutton et al.,] Sutton, R. S. et al. *Introduction to reinforcement learning*, volume 135.

[Zinkevich, 2003] Zinkevich, M. (2003). Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 928–936.