To Bid or Not to Bid in Streamlined EC2 Spot Markets

Mohan Baruwal Chhetri*, Markus Lumpe*, Quoc Bao Vo*, Ryszard Kowalczyk*†
*Faculty of Science, Engineering and Technology, Swinburne University of Technology, Melbourne, Australia
†Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland
Email: {mchhetri, mlumpe, bvo, rkowalczyk}@swin.edu.au

***Abstract*—Previously, Amazon EC2 Spot prices were always driven by *short-term trends in supply and demand*, requiring consumers to have an in-depth understanding of Spot markets and the bidding process in order to make "intelligent" *time-vs-money-vs-value trade-offs*. However, with the newly announced streamlined access model for Spot instances, Amazon states that the Spot prices will *adjust more gradually based on long-term trends* instead of reacting to short-term fluctuations in demand and supply. Therefore, consumers are no longer required to understand Spot markets and bidding, and yet can save up to 90% off the On-Demand prices. In this paper, we study the pricing patterns before and after the introduction of the modified model using standard statistical approaches including econometric inequality indices (the Gini coefficient and the Theil index), logistic regression, a hybrid forecasting technique based on *Naïve*, and Principal Component Analysis. Our findings confirm the announcements made by Amazon including less frequent Spot price changes, disappearance of sudden spikes, and smooth (but not necessarily gradual) adjustments in the Spot prices. Rather surprisingly, with the introduction of the new model, the median Spot prices have risen in the majority of the Spot markets. In addition, even in the changed access model Spot price forecasting can still yield valuable insights into the evolution and structure of a given Spot market, although there may no longer be a need for sophisticated bidding strategies.**

***Keywords*-Amazon EC2 Spot markets; Streamlined access; Defined Duration Spot Instances; Spot Prices;**

I. Introduction

Since its inception in 2006, Amazon Web Services (AWS) has been at the forefront of cloud innovation, particularly around Infrastructure as a Service (IaaS). One such innovation is the Spot Instance purchasing model which was first launched in December 2009. The central concept behind this purchasing option was *real-time supply and demand-based pricing*. Using this option, consumers could *bid the price they were willing to pay* for unused EC2 capacity and use the compute instances as long as their bid exceeded the current Spot price. The Spot price fluctuated periodically based on current demand and supply and was determined by a *sealed-bid*, *multi-unit*, *uniform price*, *market-driven auction* [1]. The primary benefit for consumers was *exact control over the maximum cost* they were willing to incur for the cloud resources, and *substantial savings* over the On-Demand prices. However, there was an associated risk of *rejection* and *revocation*. Amazon rejected Spot requests if the bid was too low and reclaimed Spot instances at any time without any notification if the current Spot price went above the customer's bid. Over time, several regulatory controls have been added to the Spot Instance purchasing option, including (a) *default limits on the maximum bid* [2], (b) *Spot Instance Termination Notice* [3], and (c) *Spot Blocks* [4].

In spite of the addition of new features, the Spot prices were always driven by short term trends in demand and supply, making it imperative for consumers to understand the Spot pricing dynamics including the presence of seasonal components, trends, and spikes when bidding for Spot Instances. In order to assist consumers with determining correct pricing levels, Amazon provided, and still does, the Spot price history for up to 90 days for all instance types across all regions.

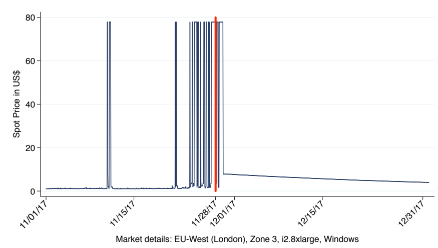


Figure 1: Amazon EC2 Spot Price Patterns - before and after the announcement of the new Spot Instance access model.

However, Amazon states that, with the introduction of a new, streamlined access model for Spot instances on November 28, 2017 [5], consumers are no longer required to have an understanding of Spot markets and bidding strategies, as the price changes are more gradual and based on longer-term trends in demand and supply, as opposed to being based on shorter-term trends in the past. Moreover, the default bid limit, which was previously 10x On-Demand price, has now been capped at 1x On-Demand price, so that the maximum Spot price that consumers may have to pay for using Spot instances is the On-Demand price. This new limit has had an immediate effect. It removes the extreme spikes that was so characteristic in the popular Spot markets [6] (cf. Fig 1).

In this changed context, several questions arise, including: (i) What is the impact of the newly streamlined access model on the evolution of Spot prices, that is, do Spot prices increase or decrease in general? (ii) Should consumers still specify maximum bid prices when requesting for Spot instances? (iii) What is the relevance (if any) of the Spot price history in estimating Spot bids going forward? (iv) Is there a need for sophisticated bidding strategies? (v) How does the regular Spot price compare with that of defined-duration Spot Block prices?

To answer these questions, we analyze the historical Spot price data for all established Windows EC2 Spot instances in the 14 generally available AWS regions for a period of 2 months starting November 1, 2017 and ending December 31, 2017, in this paper. By established Windows EC2 Spot instances we mean those that have been available as Spot instances[1] for at least 6 months prior November 1. In particular, we partition the data into two groups, using November 28, 2017 as the cut-off date to discern between the old and new access models and employ a variety of statistical tools (*i.e.*, econometric inequality analysis [7], linear and logistic regression [8], and Principal Component Analysis (PCA) [9]) to study the new Spot price patterns induced by Amazon's streamlined access model, construct statistical models for Spot price trends, and develop a forecasting technique to predict future Spot prices and the relationship between Spot prices and EC2 Spot Blocks for defined-duration workloads [4].

The results of our analyses reveal that, even in the new streamlined access model, users would still benefit from a broader understanding of the pricing trends in the Amazon EC2 ecosystem. In particular, there are still substantial savings to be had in most cases when opting for Spot instances rather than defined-duration Spot Blocks. However, the new streamlined access model clearly removes extreme spikes [6] and prices evolve more gradually at predictable, though not necessarily linear, gradients. This makes it far easier for users to successfully procure Spot instances and to estimate the associated costs. Nevertheless, there is no free lunch here either. While the absolute limit has been reduced to just the On-Demand price, we found that the median Spot price increased for 75% of the analyzed EC2 instance types in December compared to the November data. The focus of this increase was on Spot instances that would normally be traded below USD $2 per hour.

The rest of this paper is organized as follows. In Section II, we describe our statistical analysis approach and discuss a suitable bid price estimation technique that combines *linear regression* with *Naïve forecasting* [10]. We present our experimental data setup in Section III and continue with a discussion of our main observations in Section IV. We conclude with a brief discussion of related work in Section V and provide an outlook to future work in Section VI.

[1]Amazon regularly introduces new instance types to the EC2 ecosystem. These new instances only gradually give rise to spare compute capacity that is offered on the Spot market. Historically, within that phase their price was set to 10x the On-Demand price to signal that they are not intrinsically traded as Spot instances.

II. Statistical Spot Price Analysis

Statistics provides us with a rich set of methods to analyze, summarize, interpret, model, and visualize data. However, not all techniques work equally well. For example, central tendency statistics are the most popular in the domain of univariate data analysis, which aims at describing data distributions in terms of their most frequent, typical, or average values. Econometric-based methods, on the other hand, focus on the *operative differences* in data distributions [11] and, hence, yield a more effective and robust approach to reveal patterns in data [12].

In the remainder of this section, we briefly discuss all techniques used in this study and outline rationales for their utility for the purpose of Spot price data analysis.

A. Spot Price Structure Analysis

In the context of the Amazon EC2 ecosystem, *econometric inequality indices* [7] have proven particularly effective in analyzing the Spot price structure. They allow for an easy and reliable capture of two important artifacts [6]: (i) the occurrences of *Spot price extremes* that would make the market for a particular Spot instance rather disruptive, and (ii) the presence of *seasonal components* that give rise to Spot price variations due to a dependence, for example, on the *hour-of-day*. Understanding these two aspects was previously a key ingredient in developing a successful bidding strategy in the Amazon Spot market [6], [13]. We use two econometric inequality indices namely the *Gini coefficient* and the *Theil index* to analyze Spot price data. Using these two scores, Spot markets can be grouped into different categories depending upon how predictable the Spot prices are, and whether the Spot price variations depend upon the time of day.

The Gini coefficient is a statistical measure of inequality among values in a frequency distribution. For a discrete population with non-negative values x_i, $1 \leq i \leq n$, the Gini coefficient is one-half of the relative mean difference of every pair $(x_i, x_j), 1 \leq i, j \leq n$, in the population:

$$G = (1/2n^2\mu) \sum_{i=1}^{n} \sum_{j=1}^{n} |x_i - x_j|$$

The Gini coefficient takes a value within the interval $[0, 1]$ where 0 implies *perfect equality* among the values, and 1 signals *total inequality*. The Gini coefficient provides a summary measure of the price distribution for every EC2 Spot market and can be used to determine the Spot price predictability as follows:

- *Low Gini*, $[0, 0.2)$: A low Gini coefficient indicates that the Spot prices exhibit minimal deviations from the mean and are therefore very predictable.
- *Intermediate Gini*, $[0.2, 0.6)$: A Gini coefficient between 0.2 and 0.6 signals the presence of more frequent,

yet less extreme spikes and emerging seasonality in the Spot prices, making bidding more complex.

- *High Gini*, $[0.6, 1.0]$: A high Gini coefficient highlights extreme but occasional time-independent spikes in the Spot prices. Such markets can be highly predictable and yet be highly disruptive at the same time.

The Theil index is a statistical measure of inequality that supports decomposability [14] and allows, therefore, for the exploration of the structure of inequality. The Theil index takes values in the interval $[0, \log\, n]$, where a high value indicates a pronounced uneven distribution of values. For a discrete population with non-negative values x_i, $1 \leq i \leq n$, and a partitioning Π, the Theil index is given by $I(\Pi) = I_B(\Pi) + I_W(\Pi)$, with

$$I_B(\Pi) = \sum_{g=1}^{|\Pi|} \frac{X_g}{X} \log(\frac{X_g}{X} / \frac{|\Pi_g|}{n})$$

$$I_W(\Pi) = \sum_{g=1}^{|\Pi|} \frac{X_g}{X} (\sum_{i=1}^{n_g} \frac{x_{g_i}}{X_g} \log(\frac{x_{g_i}}{X_g} / \frac{1}{|\Pi_g|}))$$

where $X = \sum_{i=1}^{n} x_i$ and $X_g = \sum_{i=1}^{|\Pi|} x_{g_i}$.

The Theil index is measured as the weighted average of *inequality within subgroups* ($I_W(\Pi)$) and *inequality between subgroups* ($I_B(\Pi)$). Hence, it can be used to determine if a Spot market has recurring patterns in the Spot price variations. Technically, a Spot price history can be partitioned according to different criteria such as time of day, physical location, or instance family. Since we are interested in detecting recurring patterns in individual Spot markets over a 24 hour period, we selected the *hour-of-day* criterion.

The within-group component $I_W(\Pi)$ of the Theil index captures the variability in the Spot prices within each hour of the day, while the in-between group $I_B(\Pi)$ indicates the strength of the correlation between Spot prices and the hour of the day. The latter can be quantified by the *"in-between"* group ratio $R_B(\Pi) = I_B(\Pi)/I(\Pi)$, which can be used to explain the Spot market behavior as follows:

- *Low* $R_B(\Pi)$. This indicates the absence of seasonal components in the Spot price distribution. Therefore, bidding does not need to account for the time of day.
- *High* $R_B(\Pi)$. This conveys the presence of seasonal components in the Spot price distribution. Any successful bidding in such a market has to consider the hour of the day.

The combination of the Gini coefficient and the *"in-between"* group ratio $R_B(\Pi)$ yields an effective tool to cluster EC2 Spot markets according to their predictability [6].

B. Spot Price Change Modeling

Inequality-based analysis allows us to unearth the inherent pricing patterns in the Amazon EC2 Spot market. However, a different method is required in order to understand the impact of the newly streamlined access model on the evolution of Spot instance prices and their possible implications for successful procurement of Spot instances. A standard statistical approach to study possible relationships between variables is through *regression analysis* [8]. We are particularly interested in three aspects: (i) how have the Spot prices changed, if at all, (ii) is the change uniform across all markets, and (iii) do the December Spot prices exceed the defined-duration Spot Block prices, and if so, by what margin. To investigate these features, we employ *logistic regression*, where the independent variable is the median Spot price across all markets in December 2017, and the dependent (response) variable is a binary criterion capturing the direction of materialized price movements post introduction of the new model.

Let X be the independent variable and Y be a binary response variable. Then $\pi(x) = Pr(Y = 1|X = x)$ is the *hypothesized probability* (or odds) of an expected value x within a population of individuals satisfying $Y = 1$. A corresponding *maximum likelihood model* is given by

$$\pi(x) \quad = \quad \frac{\mathrm{e}^{(\alpha+\beta x)}}{1+\mathrm{e}^{(\alpha+\beta x)}}$$

where α is called the *intercept* and β is called the *regression coefficient*. Y is categorical with a Bernoulli distribution, where 1 represents either an increase in median Spot price or a higher median Spot price compared to the defined-duration Spot Block price, depending upon the feature being investigated. 0 otherwise.

C. Spot Price Estimation

According to Amazon, the new model is "*as easy to use as On-Demand*" [5] and does not require users to understand the Spot market landscape and use sophisticated bidding. But, how do Spot prices actually evolve? To answer this question, we developed a hybrid technique that combines *linear regression* with *Naïve* forecasting, two of the simplest techniques that have been shown to work well for many economic and financial time series [10].

For an independent variable X^l that denotes the Spot prices in the look-back period l hours and a response variable Y^l (*i.e.*, the forecast based on the look-back period), a *linear regression model* is given by

$$Y^l = \beta_0^l + \beta_1^l X^l$$

where β_0^l is the *intercept* and β_1^l is the *slope* according to look-back period l. For the purpose of Spot price forecasting, only the slope is required.[2]

To estimate Spot prices, we use the following process:

$$\mathbf{bpe}(m, d, l) = \{y_t^l(m, \hat{x}^l) : t = 1 \ldots d\}$$

[2]The intercept captures the estimate at Spot price value $x = 0$. Such Spot prices do not occur. To align the intercept with $(0, 0)$, all Spot price values have to be shifted by the mean (*i.e.*, $\forall_i \in X.x_i - \overline{x}$). The slope remains unchanged under such a transformation.

where

- m refers to the Spot market the bidder is interested in,
- $\hat{x}^l$ is last recorded/estimated Spot price,
- $d \in \{1, 6\}$ specifies the defined duration of usage in hours,
- $l \in \{28, 168, 360\}$ defines the length of the look-back period in the Spot price history in hours, and
- $y_t^l(m, \hat{x})$ is the estimated Spot price in market m based on look-back l for the t'th hour within the forecast duration d. It is given by

$$y_t^l(m, \hat{x}) = \begin{cases} \hat{x} & \text{if } \beta_1^l < 0 \\ \hat{x} + \beta_1^l \hat{x} & \text{otherwise} \end{cases}$$

  That is, we adjust the estimate based on the slope. If the slope is negative, we fall back on the last estimated Spot price, $\hat{x}$, similar to *Naïve* forecasting [10]. If the slope is positive, we estimate the new price to be $\hat{x} + \beta_1^l \hat{x}$. The rationale behind this decision is as follows. An increasing trend implies that the user has to bid at a higher estimated Spot price to secure the instance. On the other hand, with a decreasing trend the user is better off staying with the last known Spot price, as they are not penalized for inflating bids.[3]

In order to analyze the performance of our bid estimation process, we use Principal Component Analysis (PCA) [9]. PCA is an effective technique to study similarities and differences in multidimensional data sets, like the one we obtain from the established Windows EC2 Spot markets. Our estimation procedure yields 24 variables for each market to capture the respective estimation success and error rates. The associated correlation matrix has 276 coefficients. Therefore, it is essential to invest in meaningful summarization of the main relationships in a visual manner [9]. PCA allows us to draw conclusions from the linear relationships between variables by reducing the data to the principal dimensions of variability. In case of our estimation data, this means that we can bring those attributes into focus that are most relevant for the characterization of the Spot price estimation performance. In particular, with the help of PCA we are able to discern estimation performance with respect to Spot instance families and contrast them compellingly, while at the same time reducing the complexity of the data analysis.

III. Data Collection & Consolidation

We have been using the Amazon EC2 Spot price API to collect the Spot price histories of all generally available AWS regions[4] to collate an experimental data set for long-term Spot price evolution studies [6]. For the purpose of this study, we focus on all Windows instances that were active since April 2017 and used their respective Spot price histories between November 1, 2017 and December 31, 2017. We took the announcement of the new streamlined access model on November 28, 2017, to partition this selected subset into two groups. The November data solely contains Spot price histories related to the old model, whereas the December data encompasses Spot prices recorded under the new model. However, even though the new streamlined model had been announced on November 28, 2017, the data reveals that there was some latency before the new pricing process was established for an instance (cf. Fig 1).

In addition, we also collected the price histories for Defined Duration Spot Instances (for 1 and 6 hour(s)), so that we can compare standard Spot prices against Defined Duration Spot prices. Amazon does not provide any API to query the Spot Block pricing history. We, therefore, employed the *GNU Wget utility* to periodically retrieve the defined duration prices for the Windows instances.[5] We observed, while collating the data, that the last time the Defined Duration Spot prices were modified was on December 18, 2017 at 10:24 p.m. (UTC), which coincides with the opening of the 18th AWS Region in Paris [15].

The resulting data set was preprocessed as follows:

- We retained only the maximum observed Spot price for each hour, even though the Spot price can change multiple times. It is the maximum Spot price that would guarantee uninterrupted access to an instance for that particular hour.
- We divided the collected data for each instance into separate bins using the *hour-of-the-day* criterion. This is a prerequisite for calculating the Theil index and also for determining if the Spot price data contains seasonal components (*i.e.*, Spot prices change related to the time of day).
- The nature of the EC2 Spot market is such that prices can remain constant for long periods of time. As the Spot prices are only recorded when there is a change, the Spot price history may not record Spot prices for all hours. We accounted for this missing data by substituting it with the last recorded maximum Spot price so that there are no empty slots.

At the end of the preparation phase, we obtained two data sets comprising 2,553,216 records for 1,744 Windows markets across all 14 available AWS regions.

IV. Statistical Analysis Results

In the following we summarize our observations for the statistical Spot price analysis procedures outlined in Section II.

A. Spot Price Structure Summary

Figure 2 illustrates the pricing structure, in terms of the Gini coefficient and the "*in-between*" group ratio $R_B(\Pi)$,

[3] Amazon claims that Spot prices will adjust more gradually, based on longer-term trends, in the new model [5], which implies that a downward trend cannot result in a price rise through inflating bidding.

[4] The Chinese and US Government Region Endpoints are restricted.

[5] https://spot-price.s3.amazonaws.com/spotblocks-windows.js

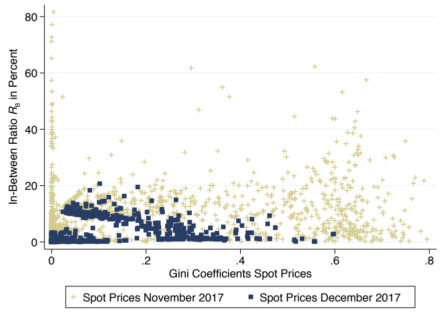


Figure 2: Amazon EC2 Price Patterns - before and after the announcement of the new Spot instance access model.

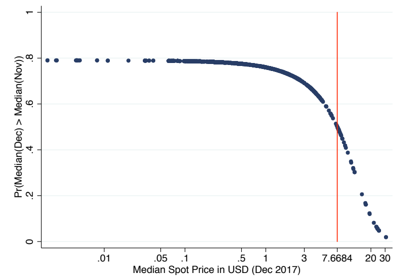


Figure 3: Maximum likelihood regression model for a higher median Spot price in the new streamlined model.

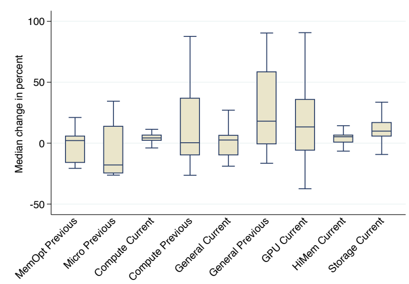


Figure 4: Median Spot price changes.

of Spot instances supporting the Windows operating system across all regions.[6] We can observe that the profiles of the pricing patterns differ in the new streamlined access model:

- *Fewer Spot price extremes.* There has been a significant drop in the Spot price extremes since the introduction of the new Spot instance access model. This fact provides evidence for a more "*gradual*" evolution of Spot prices, which are also capped at the On-Demand price now. In November 2017, there were several markets exhibiting a high Gini coefficient (between 0.6 and 0.8) due to extreme volatility in the Spot prices up to 10x the On-Demand price. In contrast, the December 2017 data shows fewer and smaller spikes, resulting in a narrower spread of the Gini coefficient for the active Windows markets. The previous Spot price volatility has almost vanished.
- *No seasonal components.* The "*hour-of-day*" has little or no bearing on the Spot prices in the new streamlined access model, as indicated by the reduced "*in-between*" group ratio $R_B(\Pi)$. In November 2017, several markets recorded strong seasonal components (*i.e.*, a dependency on the hour-of-day). This feature disappeared largely in December 2017. Spot prices do not vary based on the hour-of-day anymore in the new streamlined access model.

These two findings suggest that, in the new streamlined access model, Spot prices adjust more gradually now and are driven by long-term trends rather than random or recurring spikes and extremes.

B. Spot Price Evolution

The probability, determined by logistic regression, of median Spot prices being higher across all active Windows markets under the new streamlined model in December 2017 is shown in Fig 3.[7] The logistic model reveals that initially the odds of a higher median Spot price is 4 to 1 (*i.e.*, 80%). The odds ratio of the maximum likelihood model for a higher median Spot price is 0.8426. In other words, the odds of customers experiencing a higher median Spot price decreases with each unit of increase in the median Spot price for a given active Windows Spot instance. However, the slope of this change is very small initially, only to become significant once a median Spot price between 1 and 3 USD is reached.

Of the 1,744 studied markets, 1,305 (*i.e.*, approx. 75%) recorded higher median Spot prices in December 2017. Table I summarizes the changes per instance family. In particular, the *Compute Current* and *Storage Current* instance families account for the most frequent increases in terms of affected Spot instances. However, the actual inflation in the median Spot price varies across instance families (cf. Fig 4). The most affected markets are *Compute Previous*, *General Previous*, and *GPU Current*, where markups, when they occur, can exceed 90% compared to the median Spot prices recorded in November 2017. For *GPU Current*, we observed even higher values for individual instances (*e.g.*, US-East (N. Virginia), Zone 4&5, p2.16xlarge, 296% and US-West (Oregon), Zone 2, *g2.2xlarge*, 306%).

We used logistic regression also to study the relationships between the Spot prices and the defined-duration Spot Block prices (cf. Fig. 5). The *Spot Block Model* allows users to acquire Spot instances continuously for a fixed duration of 1 to 6 hours. This offer comes with the promise that

[6] We focus on the Windows instances only without any loss of generality. The Linux/UNIX and SUSE Linux markets exhibit similar structures.

[7] We use a logarithmic scale for the x-axis to alleviate the skewness of the median Spot prices.

	MemOpt Previous	Micro Previous	Compute Current	Compute Previous	General Current	General Previous	GPU Current	HiMem Current	Storage Current	
Higher Median Spot price	40	5	256	29	174	56	47	334	364	1,305
Lower Median Spot price	25	15	37	22	132	24	22	89	73	439
Total	65	20	293	51	306	80	69	423	437	1,744

Table I: Summary of evolution of the median Spot prices for active Windows instances in December 2017.

customers would save up to 45% compared to the On-Demand price [4]. The logistic maximum likelihood models for both, 1-hour and 6-hours duration, suggests that an actual cost saving depends on the underlying median Spot price for a given instance. More precisely, the odds of securing a Spot instance, at a price lower than its corresponding defined-duration block price, improve significantly when its median Spot price is lower than USD 6.0536 per 1 hour and USD 6.597 per 6 hours, respectively. In particular, the odds for both defined-duration offers are initially 32 to 1 (*i.e.*, 97%) against opting for defined-duration Spot Blocks, if the median Spot price is actually less than 1 cent. Only when the median Spot price for an instance approaches between 1 and 3 USD do we observe a significant upwards trend in the slope of the model function.

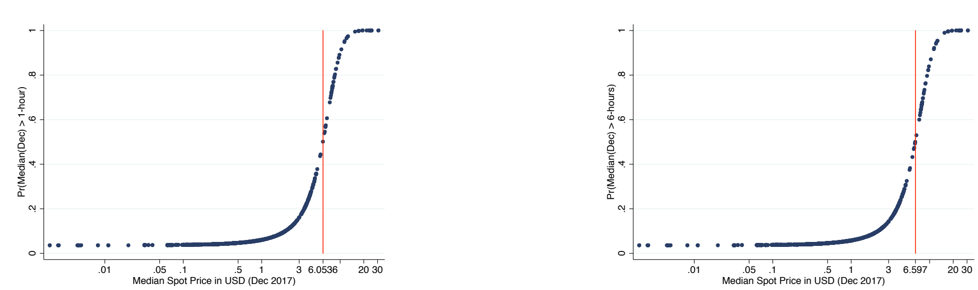
We can make the following general observations:

- The median Spot price of lower-priced Spot instances has risen (albeit by a small amount) since the new streamlined access model has been introduced.
- Consumers can secure regular Spot instances at prices lower than the defined-duration Spot Blocks for "lower-priced" Spot instances. However, with "higher-priced" Spot instances they are better off using defined-duration Spot Blocks in order to obtain the best discount.

C. Bid Price Estimation

We have previously studied various forecasting techniques and their robustness for the purpose of Spot price prediction [13]. The techniques with the best performances were *Naïve* and *Seasonal Naïve* [10]. Both approaches determine forecasts based on the last observed values. *Seasonal Naïve* outperformed *Naïve* when the underlying Spot market contained seasonal components (*i.e.*, a recurring dependency on the hour-of-day) [13]. Seasonal components have vanished in the Spot prices under the new streamlined access model. Consequently, we argue that some variation of *Naïve* forecasting should, in principle, yield good estimates for future Spot prices in the new streamlined access model.

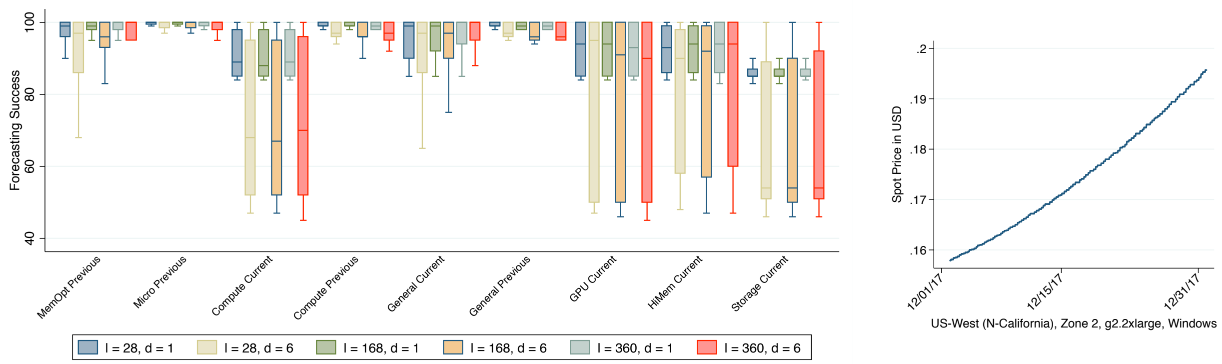
Fig 6a summarizes the success rates for the various look-back periods and defined forecast durations, as described in Section II-C, for our proposed hybrid forecasting method. We differentiate the forecasting performances via Spot instance families. We find that forecasting a block of 6 hours is, in general, rather difficult, even in the new streamlined model. On average, there is a 79% success rate when bidding for 6-hour blocks compared to 92% in case of bidding for a 1-hour block. Moreover, forecasting success is also a function of the instance family. *Compute Current*, *GPU Current*, *HiMem Current*, and *Storage Current* exhibit the highest variations in forecasting success rates, in particular, for the next 6 hours. In our technique, a forecast performs poorly if the actual Spot price grows by a factor greater than β_1^l, the slope of the linear growth model. In other words, the Spot price evolution does not follow a gradual trend line, but a step function (cf. Fig 6b). This affects long-term forecasts in particular and the instance families *Compute Current*, *GPU Current*, *HiMem Current*, and *Storage Current* especially. To



(a) 1-hour.

(b) 6-hours.

Figure 5: Maximum likelihood regression models for the median Spot price being higher than defined duration.



(a) Forecasting performance across instance families.

(b) Spot price evolution via step function.

Figure 6: Forecasting Spot prices in the new streamlined access model.

account for this discontinuous effect of Spot price evolution via a step function would possibly require the introduction of a "*step error correction*," whose value is currently unknown.

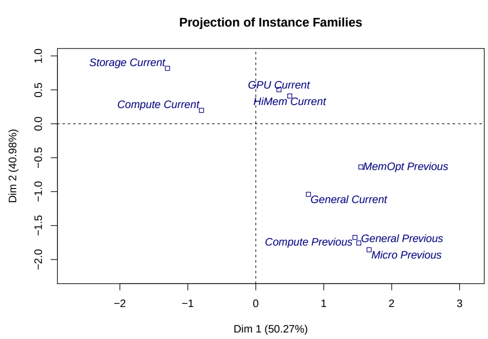


Figure 7: Amazon EC2 Spot Price - Instance Family-based Summary.

The general relationship of forecasting performance between the instance families is shown in Fig. 7, which is obtained via Principal Component Analysis (PCA). The principal plane separates the instance families according to the forecasting success and error rates. PCA maps the instance families on the first principal dimension according to the overall forecasting success. Instance families on the right enjoy a higher success rate than those on the left. The overall error rate is captured in the second principal dimensions, where the instance families mapped to the top have higher error rates. In general, we observe than the previous generation instances respond better to our forecasting method, with better success rates and lower estimation errors. The most difficult instances to forecast belong to *Storage Current* and *Compute Current*.

V. Related Work

There has been significant interest, from both academia and industry, in building "intelligent" solutions for bidding in EC2 Spot markets. Researchers have focussed on trying to understand the pricing patterns in EC2 Spot markets [1], [6], [16]. They have also focussed on developing diverse and sophisticated bidding strategies that aim at reducing costs while ensuring performance level guarantees [17]–[19]. All of these bidding strategies rely on the historical Spot pricing information [13], [20], [21] that Amazon makes available to consumers. Several commercial companies also offer services that are aimed at reducing cloud costs by taking advantage of Spot Instances including SpotInst (https://spotinst.com/), CloudCheckr (https://cloudcheckr.com/) and ParkMyCloud (https://www.parkmycloud.com/).

On the contrary, some researchers have argued against the need for sophisticated bidding strategies [22]. Their argument is that simple bidding strategies are sufficient and the focus should instead be on best-practices for deploying applications on Spot Instances. The recent introduction of the streamlined Spot instance access model, smooth price changes based on longer term trends in supply and demand, as well as support for instance hibernation [15] appears to lend support to this argument, as these features aim at a more gradual change of Spot prices [5], possibly removing the need to closely follow the Spot price changes. The actual evolution of Spot prices is more complex though. The December 2017 data reveals that Spot prices can change via a step function faster than the underlying linear gradient.

The recently announced features indicate that Amazon aims at simplifying the use of Spot instances and our initial experimental results confirm this to some degree. Spot price prediction using simple *linear regression* combined with *Naïve* forecasting, as discussed in this work, can produce estimates for Spot prices with a reasonable level of accuracy, if the underlying Spot pricing patterns adhere to a gradual change profile. However, while our current results are based on the analysis of the new Spot prices for just one month, we will have a clearer understanding of the impact of the new pricing model when we have more data to analyze.

VI. Conclusion

In this paper, we have presented the results of our analysis of the EC2 Spot markets since the new streamlined access model with smooth price changes was announced. We used standard statistical measures including econometric inequality indices (the Gini coefficient and the Theil index), logistic regression, a hybrid forecasting technique based on *Naïve*, and PCA to study the pricing patterns before and after the introduction of the modified model.

Our findings confirm the announcements made by Amazon regarding the price changes in the Spot markets – changes are less frequent, sudden spikes have disappeared, and changes entail smooth adjustments, though not necessarily gradual. Rather surprisingly, however, we also discovered that with the introduction of the new model the median Spot prices have risen in the majority of the Spot markets. The actual inflation may be moderate, but the exact reasons for this effect are unknown.

Spot price forecasting can still yield valuable insights into the evolution and structure of a given Spot market. However, the accuracy of estimates depends heavily on the presence of a step function in the evolution of Spot prices. Therefore, in the future we seek to explore this aspect more in an attempt to account for discontinuous Spot prices in forecasting.

Acknowledgment

This research was supported by the Australian Research Council's Linkage Projects funding scheme (Project No. LP150100846) on *Consumer-centric Adaptive Quality-assured Cloud Services Brokerage*.

References

[1] O. Agmon Ben-Yehuda, M. Ben-Yehuda, A. Schuster, and D. Tsafrir, "Deconstructing Amazon EC2 spot instance pricing," *ACM Transactions on Economics and Computation*, vol. 1, no. 3, p. 16, 2013.

[2] K. Simpson, "Amazon Web Services Puts Cap On Spot Instance Pricing." [Online]. Available: https://blog.mailchannels.com/2013/11/aws-puts-a-cap-on-spot-instance-pricing

[3] J. Barr, "New - EC2 Spot Instance Termination Notices," Jan 2015. [Online]. Available: https://aws.amazon.com/blogs/aws/new-ec2-spot-instance-termination-notices/

[4] ——, "New EC2 Spot Blocks for Defined-Duration Workloads," Oct 2015. [Online]. Available: https://aws.amazon.com/blogs/aws/new-ec2-spot-blocks-for-defined-duration-workloads/

[5] ——, "Amazon EC2 Update Streamlined Access to Spot Capacity, Smooth Price Changes, Instance Hibernation," Nov 2017. [Online]. Available: https://aws.amazon.com/blogs/aws/amazon-ec2-update-streamlined-access-to-spot-capacity-smooth-price-changes-instance-hibernation/

[6] M. Lumpe, M. Baruwal Chhetri, Q. B. Vo, and R. Kowalczyk, "On Estimating Minimum Bids for Amazon EC2 Spot Instances," in *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. IEEE Press, 2017, pp. 391–400.

[7] A. Sen, *On economic inequality*. Oxford University Press, 1973.

[8] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2013.

[9] F. Husson, S. Lê, and J. Pagès, *Exploratory multivariate analysis by example using R*. CRC press, 2017.

[10] R. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*. OTexts.com, 2014.

[11] O. Goloshchapova and M. Lumpe, "On the application of inequality indices in comparative software analysis," in *22nd Australian Software Engineering Conference (ASWEC 2013)*. IEEE, 2013, pp. 117–126.

[12] R. Vasa, M. Lumpe, P. Branch, and O. Nierstrasz, "Comparative Analysis of Evolving Software Systems Using the Gini Coefficient," in *Proceedings of 25th IEEE International Conference on Software Maintenance (ICSM '09)*. Edmonton, Alberta: IEEE Computer Society, Sep. 2009, pp. 179–188.

[13] M. Baruwal Chhetri, M. Lumpe, Q. B. Vo, and R. Kowalczyk, "On estimating bids for amazon ec2 spot instances using time series forecasting," in *Services Computing (SCC), 2017 IEEE International Conference on*. IEEE, 2017, pp. 44–51.

[14] H. Theil, "A New Index Number Formula," *The Review of Economics and Statistics*, vol. 55, no. 4, pp. 498–502, Nov. 1973.

[15] J. Barr, "Now Open AWS EU (Paris) Region," Dec 2017. [Online]. Available: https://aws.amazon.com/blogs/aws/now-open-aws-eu-paris-region/

[16] B. Javadi, R. K. Thulasiram, and R. Buyya, "Characterizing spot price dynamics in public cloud environments," *Future Generation Computer Systems*, vol. 29, no. 4, pp. 988–999, 2013.

[17] L. Zheng, C. Joe-Wong, C. W. Tan, M. Chiang, and X. Wang, "How to bid the cloud," *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 4, pp. 71–84, 2015.

[18] M. Zafer, Y. Song, and K.-W. Lee, "Optimal bids for spot vms in a cloud for deadline constrained jobs," in *5th IEEE International Conference onCloud Computing (CLOUD 2012)*. IEEE, 2012, pp. 75–82.

[19] Y. Song, M. Zafer, and K.-W. Lee, "Optimal bidding in spot instance market," in *31st Annual IEEE International Conference on Computer Communications (INFOCOM 2012)*. IEEE, 2012, pp. 190–198.

[20] M. Baruwal Chhetri, M. Lumpe, Q. B. Vo, and R. Kowalczyk, "On forecasting amazon ec2 spot prices using time-series decomposition with hybrid look-backs," in *Edge Computing (EDGE), 2017 IEEE International Conference on*. IEEE, 2017, pp. 158–165.

[21] R. Wolski, J. Brevik, R. Chard, and K. Chard, "Probabilistic guarantees of execution duration for amazon spot instances," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. ACM, 2017, p. 18.

[22] P. Sharma, D. Irwin, and P. Shenoy, "Keep it simple: Bidding for servers in today's cloud platforms," *IEEE Internet Computing*, vol. 21, no. 3, pp. 88–92, 2017.