Soft Computing (2023) 27:13677–13688
https://doi.org/10.1007/s00500-022-07404-1

FOCUS

Penalized logistic regressions with technical indicators predict up and down trends

Huifeng Jiang[1] · Xuemei Hu[2] · Hong Jia[1]

Accepted: 27 May 2022 / Published online: 16 August 2022

Abstract

Correctly predicting up and down trends for stock prices is of immense important in the financial market. To further improve the prediction performance, in this paper we introduce five penalties: ridge, least absolute shrinkage and selection operator, elastic net, smoothly clipped absolute deviation and minimax concave penalty to logistic regressions with 19 technical indicators, and propose the five penalized logistic regressions to predict up and down trends for stock prices. Firstly, we translate the five penalized logistic log-likelihood functions into the five penalized weighted least squares functions and combine them with the tenfold cross-validation method to calculate the solution path to parameter estimators. Secondly, we combine the binomial deviation with cross-validation error as a risk measure to choose an appropriate tuning parameter for the penalty functions and apply the training set and the coordinate descent algorithm to obtain parameter estimators and probability estimators. Thirdly, we employ the testing set and the chosen optimal thresholds to construct two-class confusion matrices and receiver operating characteristic curves to assess the prediction performances to the five regressions. Finally, we compare the proposed five penalized logistic regressions with logistic regression, support vector machine and artificial neural network and found that the minimax concave penalty logistic regression performs the best in terms of the prediction performance to up and down trends for Google's stock prices. Therefore, in this paper we propose the five new prediction methods to improve the prediction accuracy of stock returns and bring economic benefits for investors.

Keywords Penalized logistic regressions · Up and down trends · Coordinate descent algorithm · Support vector machine · Artificial neural network

1 Introduction

Stock market exists some inherent characteristics such as model uncertainty, parameter instability and noise accumulation. These characteristics make the stock market prediction more complex. Different viewpoints spring up in economic and finance. For example, both efficient market hypothesis and random walk theory assumed that the stock market was unpredictable, whereas Dow theory and Murphy (1999) assumed that financial market was predictable. In particular, Murphy (1999) proposed many technical indicators and developed the technical analysis methods for finance market, whereas Elliott et al. (2013) systematically summarized the economic forecasting problems, emphasized the challenges from stock price forecasting and provided the strategies to improve the forecasting performances. In recent years, some machine learning methods have been proposed to predict stock market. For example, Wang and Zhu (2010) developed support vector regression and a two-step kernel learning method for financial time series prediction. Nair et al. (2011) proposed adaptive artificial neural network (ANN) to predict the second-day closing price of stock market index. Cavalcante et al. (2016) systematically reviewed the progress

Communicated by Priti Bansal.

Xuemei Hu
huxuem@163.com

Huifeng Jiang
jianghuifeng0221@163.com

Hong Jia
jh9829@ctbu.edu.cn

[1] Research Center for Economy of Upper Reaches of the Yangtse River, Chongqing Technology and Business University, 19 Xuefu Avenue, Chongqing 400067, China

[2] School of Mathematics and Statistics, Chongqing Key Laboratory of Social Economy and Applied Statistics, Chongqing Technology and Business University, 19 Xuefu Avenue, Chongqing 400067, China

on artificial intelligence, neural network and support vector machine (SVM) in predicting the change of stock price or direction. Zhang et al. (2018) proposed a novel stock price trend prediction system that could predict both stock price movement and its interval of growth (or decline) rate within the predefined prediction durations. Wen et al. (2019) introduced a new method to simplify noisy-filled financial temporal series via sequence reconstruction by leveraging motifs (frequent patterns) and then utilized a convolutional neural network to predict up and down trends for stock prices. Nabipour et al. (2020) applied machine learning and deep learning algorithms to significantly reduce the risk of trend prediction. Shen and Shafiq (2020) proposed a comprehensive customization of feature engineering and deep learning-based model to predict price trends for China's stock markets.

It is well known that public sentiment is closely linked to financial markets. In recent years, the impact of investor sentiment on stock returns has been investigated. For example, Joshi et al. (2016) predicted the future stock movements through news sentiment classification. Li et al. (2017) proposed a long short-term memory neural network by combining investor sentiment with market factors to improve the prediction performance. Xing et al. (2019) proposed a novel sentiment-aware volatility forecasting model to produce the more accurate estimation for temporal variances to asset returns by capturing the bi-directional interaction between movements of asset price and market sentiment. Khan et al. (2020) proposed machine learning methods with sentiment and situational features to predict future movements of stocks. Li et al. (2021) constructed the return distributions for the Shanghai Security Composite Index by adding sentiment-aware variables. In addition, market sentiment perspectives and public sentiment-driven portfolio or asset allocation has been also analyzed. For example, Malandri et al. (2018) discussed how the public sentiment would affect portfolio management. Xing et al. (2018) investigated the role of market sentiment in an asset allocation problem. Xing et al. (2018) proposed to formalize public sentiment as a market views and integrated it into modern portfolio theory. Picasso et al. (2019) combined technical analysis with sentiment analysis for news and constructed a portfolio return forecasting model by machine learning, etc..

Predicting up and down trends for stock prices is an important puzzle in the financial field. Even very small improvements in the prediction performance can be very profitable. For example, Hu and Jiang (2021) proposed logistic regression with 6 technical indicators to predict up and down trends for Google's stock prices and obtain the higher prediction accuracy. In this paper we introduce the five penalties: ridge, least absolute shrinkage and selection operator (LASSO), elastic net, smoothly clipped absolute deviation (SCAD) and minimax concave penalty (MCP) to logistic regressions with 19 technical indicators, and propose the five penalized logistic regressions to further improve the prediction performance to stock returns. Firstly, we combine the iterative weighted least squares algorithm with the tenfold cross-validation method, calculate the overall solution path of model parameters and select a specific solution path from the overall solution path. Secondly, we combine the binomial deviation with cross-validation error as a risk measure to choose an appropriate tuning parameter λ and apply the training set and the coordinate descent algorithm to obtain parameter estimators and probability estimators. Thirdly, we employ the testing set and the chosen optimal thresholds to construct two-class confusion matrices and receiver operating characteristic (ROC) curves to assess the prediction performances to the five regressions. Finally, we compare the proposed five penalized logistic regressions with logistic regression, SVM and ANN, and found that the MCP logistic regression performs the best in terms of the prediction performance to stock returns. So we recommend investors to employ the MCP logistic regression to predict up and down trends for stock prices and gain the richer economic benefit.

The rest of this paper is organized as follows. In Sect. 2, we establish the five penalized logistic regressions with technical indicators. In Sect. 3, we apply the training set to learn the five penalized logistic regressions and obtain parameter estimators and probability estimators. In Sect. 4, we adopt the testing set to obtain two class confusion matrices and ROC curves for the five regressions to assess their prediction performances. In Sect. 5, we compare the proposed five prediction methods with logistic regression, SVM and ANN.

2 Penalized logistic regressions

Let C_t be the closing price of a given stock at the end of the t-th trading day, $K_t = C_{t+1} - C_t$ be the stock excess return,

$$Y_t = \begin{cases} 1, & \text{if} \quad K_t > 0, \\ 0, & \text{if} \quad K_t \leq 0, \end{cases} \tag{1}$$

represents the direction indicator function, where $Y_t = 1$ represents up trends, and $Y_t = 0$ represents down trends. The main goal of this paper is to predict up and down trends for stock prices. In the following we apply a training set $D = \{x_t, y_t\}_{t=1}^n$ to learn up and down trends for stock prices and construct a two-category classification rule that may be hidden deeply in the raw dataset, where x_t is the sample from the predictor vector X_t whose distribution is usually unknown. It is well-known that logistic regression is a powerful two-category classification method. In this paper we combine logistic regression with technical analysis developed by Murphy (1999) and proposed the following logistic regression with 19 technical indicators:

$$
\begin{aligned}
&P\left(X_t;\beta_0,\beta\right)=P\left(Y_t=1 \mid X_t;\beta_0,\beta\right)\\
&=\frac{\exp\left(\beta_0+X_t^{\top}\beta\right)}{1+\exp\left(\beta_0+X_t^{\top}\beta\right)},
\end{aligned}
\tag{2}
$$

$$
\begin{aligned}
&1-P\left(X_t;\beta_0,\beta\right)=P\left(Y_t=0 \mid X_t;\beta_0,\beta\right)\\
&=\frac{1}{1+\exp\left(\beta_0+X_t^{\top}\beta\right)},
\end{aligned}
\tag{3}
$$

where β_0 is an unknown intercept term, $\beta = (\beta_1, \beta_2, \ldots, \beta_{19})^{\top}$ is an unknown parameter vector, and $X_t = (X_{t,1}, X_{t,2}, \ldots, X_{t,19})^{\top}$ is the predictor vector composed of 19 technical indicators listed in Table 1. To avoid multi-collinearity and over-fitting, we introduce the five penalties for logistic regression to remove some technical indicators that are irrelevant to up and down trends for stock prices and construct the five penalized logistic regressions to predict up and down trends for stock prices. Let $x_t = (x_{t,1}, x_{t,2}, \ldots, x_{t,19})^{\top}$ and y_t be the observation samples for X_t and Y_t, respectively. Given the training set $\{x_t, y_t\}_{t=1}^{n}$, we obtain the following negative log-likelihood

$$
\begin{aligned}
l(\beta)=-L(\beta)=-\sum_{t=1}^{n}\Big\{&y_t\left(\beta_0+x_t^{\top}\beta\right)\\
&-\log\left[1+\exp\left(\beta_0+x_t^{\top}\beta\right)\right]\Big\},
\end{aligned}
\tag{4}
$$

and the penalized negative log-likelihood function

$$
Q(\beta;\lambda,\gamma)\equiv l(\beta)+p_{\lambda,\gamma}(\beta),
\tag{5}
$$

where $p_{\lambda,\gamma}(\beta)$ is a function of the coefficients indexed by a tuning parameter λ that controls the trade-off between the loss function and penalty, and that also may be shaped by one or more regularization parameters γ. In this paper we choose the five penalty functions listed in Table 2.

3 Parameter estimators and probability estimators

Negative log-likelihood function (4) is not differentiable. Hence if the current estimates of the parameters are $(\widehat{\beta}_0, \widehat{\beta}(m))$, we transform (4) into the weighted least-squares function and form a quadratic approximation to negative log-likelihood function (4):

$$
\begin{aligned}
l_Q(\beta_0,\beta)=-\frac{1}{2n}\sum_{t=1}^{n}&W_t(\widetilde{Y}_t-\beta_0-x_t^{\top}\beta)^2\\
&+C(\widehat{\beta}_0,\widehat{\beta}(m))^2,
\end{aligned}
\tag{6}
$$

where

$$
\widetilde{Y}_t=\widehat{\beta}_0+x_t^{\top}\widehat{\beta}(m)+\frac{y_t-\widetilde{P}_t}{\widetilde{P}_t(1-\widetilde{P}_t)},
$$

the estimator $\widetilde{P}_t = \frac{\exp(x_t^{\top}\widehat{\beta}(m))}{1+\exp(x_t^{\top}\ \ \widehat{\beta}(m))}$, of P_t add the estimator $\widehat{\beta}_0$ of the intercept β_0 as follows:

$$
W_t=\widetilde{P}_t(1-\widetilde{P}_t),\ \widetilde{P}_t=\frac{\exp\left(\widehat{\beta}_0+x_t^{\top}\widehat{\beta}(m)\right)}{1+\exp\left(\widehat{\beta}_0+x_t^{\top}\ \ \widehat{\beta}(m)\right)}
\tag{7}
$$

and $C(\widehat{\beta}_0, \widehat{\beta}(m))^2$ is constant. Similarly, penalized negative log-likelihood function (5) is not differentiable. Therefore, we replace the negative log-likelihood function $l(\beta)$ in (5) by the weighted least-squares function $l_Q(\beta_0, \beta)$, run the coordinate descent algorithm to obtain the parameter estimator

$$
\widehat{\beta}^{\lambda,\gamma}=\arg\min_{\beta}\left\{l_Q(\beta_0,\beta)+p_{\lambda,\gamma}(\beta)\right\},
\tag{8}
$$

where the intercept term β_0 does not be penalized. More details refer to Breheny and Huang (2011) on the coordinate descent algorithm for penalized logistic regressions. Table 3 lists three specific parameter estimators.

For j in $\{1, 2, \ldots, p\}$, the coordinate descent algorithm partially optimizes a target function $Q(\beta; \lambda, \gamma)$ with respect to a single parameter β_j with the remaining parameters $\beta_l, l \neq j$ fixed at their most recently updated values $\widehat{\beta}_1^{\lambda,\gamma}(m+1), \ldots, \widehat{\beta}_{j-1}^{\lambda,\gamma}(m+1), \widehat{\beta}_{j+1}^{\lambda,\gamma}(m), \ldots, \widehat{\beta}_p^{\lambda,\gamma}(m)$, then iteratively cycling through all the parameters until convergence or a maximum iteration number M is reached, and this process repeats over a grid of values for λ to produce a path of the solution. Usually, we are interested in obtaining $\widehat{\beta}^{\lambda,\gamma}$ not just for a single value of $\lambda \in [\lambda_{min}, \lambda_{max}]$, but for a range of values extending from a maximum value λ_{max} for which all penalized coefficients are 0 down to $\lambda = 0$ or to a minimum value λ_{min} at which the model becomes excessively large or ceases to be identifiable. Thus, by starting at λ max with $\beta(0) = 0$ and proceeding toward λ_{min}, we can ensure that the initial values will never be far from the solution. For γ, we generally take $\gamma = 3.7$. Here we take the different values for γ and found that $\gamma = 5$ for MCP and $\gamma = 10$ for SCAD are better. Algorithm 1 provides the specific pseudocode on how to apply the coordinate descent algorithm to calculate the parameter estimators for the MCP logistic regression. The coordinate descent algorithms to parameter estimators for the other four penalized logistic regressions are similar to Algorithm 1. We would not list them here for lack of space.

In this paper we apply the coordinate descent algorithm to the five penalized logistic regressions to obtain the final

Table 1 Nineteen technical indicators and their formulae

Indicators	Descriptions	Formulae
$X_{t,1}$(WMA)	Weighted moving average	$WMA_t = [nP_t + (n-1)P_{t-1} + \ldots + P_1]/n!.$
$X_{t,2}$(DEMA)	Double exponential moving average	$DEMA_t(n) = 2EMA_t(n) - EMA_t(EMA_t(n)),$ $EMA_t(n) = [2P_t + (n-1)EMA_{t-1}(n)]/(n+1).$
$X_{t,3}$(ADX)	Average directional movement Index measures the strength of a trend	$ADX_t = [(n-1)ADX_{t-1} + DX_t]/n,$ $DX_t = [(+DI_t) - (-DI_t)]/[(+DI_t) + (-DI_t)],$ $+DI_t = H_t - H_{t-1}, -DI_t = L_{t-1} - L_t\ .$
$X_{t,4}$(MACD)	Moving average convergence divergence compares a fast exponential moving average with a slow exponential moving average	$MACD_t = EMA_t(s) - EMA_t(t), s < t.$
$X_{t,5}$(CCI)	Commodity channel index measures the current price relative to an average price	$CCI_t = (M_t - SM_t)/0.015D_t,$ $M_t = (H_t + L_t + C_t)/3, SM_t = \sum_{i=1}^{n} M_{t-i+1}/n,$ $D_t = \sum_{i=1}^{n} \mid M_{t-i+1} - SM_t \mid /n.$
$X_{t,6}$(MO)	Momentum provides the difference of a series over two observations	$MO_t(k) = P_t - P_{t-k}.$
$X_{t,7}$(RSI)	Relative strength index measures velocity magnitude of directional price movements	$RSI_t(n) = 100 - 100/[1 + RS_t(n)],$ $RS_t(n) = UP_{avg}(n)/DOWN_{avg}(n).$
$X_{t,8}$(ATR)	Average true range	$TR_t = \text{Max}\,[(H_t - L_t)\,, (H_t - C_t)\,, (L_t - C_t)],$ $ATR_{\mathrm{t}}(n) = \frac{1}{n} \sum_{t=1}^{n} TR_t.$
$X_{t,9}$(CLV)	Close location value is a metric utilized in technical analysis to assess where the closing price of a security falls relative to its day’s high and low prices	$CLV_t = \frac{C_t - L_t - (H_t - C_t)}{H_t - L_t}.$
$X_{t,10}$(CMF)	Chaiken money flow compares the whole volume with regard to the close, high and low prices.	$CLV_t = [(C_t - L_t) - (H_t - C_t)]/(H_t - C_t),$ $CMF_t = \sum(CLV_t \times VO_t)/\sum VO_t.$
$X_{t,11}$(CMO)	Chande momentum oscillator	$CMO_t = \frac{SU_t - SD_t}{SU_t + SD_t} \times 100.$
$X_{t,12}$(EMV)	Ease of movement value	$BR_t = \frac{V_t}{H_t - L_t}, EMV_t = \frac{MPM_t}{BR_t},$ $MPM_t = \left(\frac{H_t + L_t}{2}\right) - \left(\frac{H_{t-1} + L_{t-1}}{2}\right).$
$X_{t,13}$(MFI)	Money flow index uses price and volume data for identifying overbought or oversold signals in an asset.	$TP_t = \frac{H_t + L_t + C_t}{3}, RMF_t = TP_t \times V_t,$ $MFR_t = \frac{14PPMF_t}{14PNMF_t}, MFI_t = 100 - \frac{100}{1 + MFR_t}.$
$X_{t,14}$(ROC)	Rate of change	$ROC_t = C_t/C_{t-n} \times 100.$
$X_{t,15}$(VHF)	Vertical horizontal filter	

Table 1 continued

Indicators	Descriptions	Formulae
	can distinguish the types of market.	$VHF_t = \frac{HCP_t - LCP_t}{\sum \lvert C_{t-i+1} - C_{t-i} \rvert}.$
$X_{t,16}$(SAR)	Parabolic stop-and-reverse is used to determine the direction of a trend and the potential reversal of a price	$SAR_t = SAR_{t-1} + AF\,(H_{t-1} - SAR_{t-1})$ or $SAR_t = SAR_{t-1} + AF\,(L_{t-1} - SAR_{t-1}).$
$X_{t,17}$(TRIX)	Triple smoothed exponential oscillator is to filter price noise and insignificant price movements	$TR_t(n) = EMA\left(EMA\left(EMA\left(C_t, n\right), n\right), n\right),$ $TRIX_t(n) = 100 \times (TR_t(n)/TR_{t-1}(n) - 1).$
$X_{t,18}$(WPR)	William's indicator is a dynamic technical indicator that determines whether the Market is overbought or bought.	$WPR_t = (H_{t-n} - C_t)\,/\,(H_{t-n} - L_{t-n}) \times 100.$
$X_{t,19}$(SNR)	Signal to noise ratio can see the trend direction of the stock	$SNR_t = \lvert\, C_t - C_{t-n} \,\rvert\, / ATR_n.$

Table 2 Penalized functions

Penalties	Formulae
Ridge	$p_{\lambda(\beta)} = \lambda \lVert\beta\rVert_2^2.$
LASSO	$p_{\lambda}(\beta) = \lambda \lVert\beta\rVert_1.$
ENet	$p_{\lambda,\gamma}(\beta) = \frac{1}{2}\lambda\left[(1-\gamma)\lVert\beta\rVert_2^2 + \gamma\lVert\beta\rVert_1\right], \lambda \in (0, \infty), \gamma \in (0, 1).$
MCP	$p_{\lambda,\gamma}(\beta) = \begin{cases} \lambda\beta - \frac{\beta^2}{2\gamma}, & \text{if } \beta \leq \gamma\lambda, \\ \frac{1}{2}\gamma\lambda^2, & \text{if } \beta > \gamma\lambda, \end{cases} \quad \lambda \geq 0\,, \gamma > 1.$
SCAD	$p_{\lambda,\gamma}(\beta) = \begin{cases} \lambda\beta, & \text{if } \beta \leq \lambda, \\ \frac{\lambda\gamma\beta - 0.5(\beta^2+\lambda^2)}{(\gamma-1)}, & \text{if } \lambda < \beta \leq \lambda\gamma, \\ \frac{\lambda^2(\gamma+1)}{2}, & \text{if } \beta > \lambda\gamma. \end{cases} \quad \lambda \geq 0, \gamma > 2.$

ENet represents elastic net

Table 3 Penalized functions and parameter estimators for penalized logistic regressions

Penalties	Estimators
LASSO	$\widehat{\beta}_j^{LASSO}\left(Z_j; \lambda\right) = \frac{S(Z_j, \lambda)}{\nu_j}.$
MCP	$\widehat{\beta}_j^{MCP}\left(Z_j; \lambda, \gamma\right) = \begin{cases} \frac{S(Z_j, \lambda)}{\nu_j - 1/\gamma}, & \lvert Z_j \rvert \leq \nu_j\lambda\gamma, \\ \frac{Z_j}{\nu_j}, & \lvert Z_j \rvert > \nu_j\lambda\gamma, \end{cases} \quad \gamma > 1/\nu_j.$
SCAD	$\widehat{\beta}_j^{SCAD}\left(Z_j; \lambda, \gamma\right) = \begin{cases} \frac{S(Z_j, \lambda)}{\nu_j}, & \lvert Z_j \rvert \leq \lambda\left(\nu_j + 1\right), \\ \frac{S(Z_j, \gamma\lambda/(\gamma-1))}{\nu_j - 1/(\gamma-1)}, & \lambda\left(\nu_j + 1\right) < \lvert Z_j \rvert \leq \nu_j\lambda\gamma, \gamma > 1 + 1/\nu_j. \\ \frac{Z_j}{\nu_j}, & \lvert Z_j \rvert > \nu_j\lambda\gamma, \end{cases}$
Symbols	$\widehat{P}_t = \exp(\widehat{\beta}_0^{\lambda,\gamma} + x_t^\top \widehat{\beta}^{\lambda,\gamma}(m))/[1 + \exp(\widehat{\beta}_0^{\lambda,\gamma} + x_t^\top \widehat{\beta}^{\lambda,\gamma}(m)], W_t = \widehat{P}_t(1 - \widehat{P}_t), t = 1, \ldots, n,$ $W = diag\{W_1, W_2, \ldots, W_n\}, \widetilde{Y} = x^\top \widehat{\beta}^{\lambda,\gamma}(m) + W^{-1}(Y - \widehat{P}), \widehat{P} = (\widehat{P}_1, \ldots, \widehat{P}_n),$ $x_{\cdot j} = (x_{1j}, \ldots, x_{nj})^\top, \nu_j = n^{-1} x_{\cdot j}^\top W x_{\cdot j}, j = 1, \cdots, p,$ $Z_j = n^{-1} x_{\cdot j}^\top W\left(\widetilde{Y} - x_{\cdot -j}\beta_{-j}\right) = n^{-1} x_{\cdot j}^\top W r + \nu_j \widehat{\beta}_j^{\lambda,\gamma}(m),$ $x_{\cdot -j} = \left(x_{\cdot 1}, \cdots, x_{\cdot j-1}, 0, x_{\cdot j+1}, \cdots, x_{\cdot p}\right)^\top,$ $\beta_{-j} = \left(\beta_1, \cdots, \beta_{j-1}, 0, \beta_{j+1}, \cdots, \beta_p\right).$

parameter estimators $\widehat{\beta}_0^{\lambda,\gamma}$ and $\widehat{\beta}^{\lambda,\gamma}$, then compute the probability estimators

$$\widehat{P}\left(Y_t = 1 \mid X_t; \widehat{\beta}_0^{\lambda,\gamma}, \widehat{\beta}^{\lambda,\gamma}\right) = \frac{\exp\left(\widehat{\beta}_0^{\lambda,\gamma} + X_t^\top \widehat{\beta}^{\lambda,\gamma}\right)}{1 + \exp\left(\widehat{\beta}_0^{\lambda,\gamma} + X_t^\top \widehat{\beta}^{\lambda,\gamma}\right)}, \tag{9}$$

$$\widehat{P}\left(Y_t = 0 \mid X_t; \widehat{\beta}_0^{\lambda,\gamma}, \widehat{\beta}^{\lambda,\gamma}\right) = \frac{1}{1 + \exp\left(\widehat{\beta}_0^{\lambda,\gamma} + X_t^\top \widehat{\beta}^{\lambda,\gamma}\right)}. \tag{10}$$

Remark Compared with local linear/quadratic approximation algorithm, the coordinate descent algorithm has the following advantages: 1) The optimization over each single parameter has a single closed solution; 2) updating can be computed very rapidly; 3) initial values will never be far from the solutions and a few iterations are required.

Algorithm 1 Coordinate descent for MCP logistic regression

```
Require: the training set $\left\{x_t = \left(x_{t,1}, x_{t,2}, \cdots, x_{t,p}\right), y_t\right\}_{t=1}^n$,a grid of
    increasing $\lambda$ values $\Lambda = \{\lambda_1, \ldots, \lambda_L\}$, $\gamma = 5$, a given tolerance
    limit $\varepsilon$ and a maximum iteration number M
Initialization $\widehat{\beta}(0) = \widehat{\beta}\,(\lambda_{max} = \lambda_L, \gamma = 5)$
for each $m = 0, 1\cdots$, each $l \in \{L, L-1, \cdots, 1\}$, do
    repeat
        $\widehat{\eta}_t \Leftarrow \beta_0 + x_t^\top \widehat{\beta}^{\lambda,\gamma}(m)$
        $\widehat{P}_t \Leftarrow \left\{e^{\widehat{\eta}_t} / \left(1 + e^{\widehat{\eta}_t}\right)\right\}_{t=1}^n$
        $W \Leftarrow \mathrm{dig}\left\{\widehat{P}_1\left(1 - \widehat{P}_1\right), \cdots, \widehat{P}_n\left(1 - \widehat{P}_n\right)\right\}$
        $r \Leftarrow W^{-1}\{Y - \widehat{P}\}$
        $\widetilde{Y} \Leftarrow \eta + r$
        while not convergent do
            for each $j \in \{1, 2, \cdots, p\}$ do
                $\nu_j \Leftarrow n^{-1} x_{\cdot j}^\top W x_{\cdot j}$
                $Z_j \Leftarrow \frac{1}{n} x_{\cdot j}^\top W \left(\widetilde{Y} - x_{\cdot -j}\beta_{-j}\right)$
                    $\Leftarrow \frac{1}{n} x_{\cdot j}^\top W r + \nu_j \widehat{\beta}_j^{\lambda,\gamma}(m)$
                where set the $\lambda$ for the intercept term to 0
                if $|Z_j| \le \nu_j \gamma \lambda$ then
                    $\widehat{\beta}_j^{\lambda,\gamma}(m+1) \Leftarrow \frac{S(Z_j,\lambda)}{\nu_j - 1/\gamma}$
                else
                    $\widehat{\beta}_j^{\lambda,\gamma}(m+1) \Longleftarrow \frac{Z_j}{\nu_j}$
                end if
                $r \Leftarrow r - x_{\cdot j}^\top \left(\widehat{\beta}_j^{\lambda,\gamma}(m+1) - \widehat{\beta}_j^{\lambda,\gamma}(m)\right)$
            end for
        end while
    until $\left\|\widehat{\beta}^{\lambda,\gamma}(m+1) - \widehat{\beta}^{\lambda,\gamma}(m)\right\|_2^2 \le \varepsilon$ or do a maximum itera-
    tion number M
end for
Ensure: $\widehat{\beta}^{\lambda,\gamma}$
```

4 Two-class prediction performance

Two-class confusion matrix is a contingency table of the true class and the predicted class that describes two-class classification results, see Table 4.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{11}$$

that is the simplest index to evaluate the prediction performance. However, it cannot reflect the losses from two types of errors. Therefore, a ROC curve is introduced to evaluate the prediction performance. Suppose that $TPR(c) = P(X_1 < c)$ represents the true positive rate at the threshold c, and $FPR(c) = P(X_2 < c)$ represents the false positive rate at the threshold c. By setting the different threshold c, we calculate $\{(TPR(c), FPR(c))\}$ or (Sensitivity, 1-Specificity) to draw a ROC curve, where

$$\text{Sensitivity(True positive rate,TPR)} = TP/(TP + FN), \tag{12}$$

$$\text{Specificity(1-False positive rate,1-FPR)} = TN/(TN + FP). \tag{13}$$

In Sect. 5 we adopt the R package pROC to draw a ROC curve and compute AUC (the area under the ROC curve, a summary indicator of classification performance). More details on ROC can refer to Chapter 7 in Hu and Liu (2020).

5 Real data analysis

5.1 Technical indicators and variance inflation factors

The stock market fluctuates greatly during December 2019 because of the novel coronavirus pandemic. Therefore we select Google's stock prices from January 2010 to November 2019 as the observation data with the sample size $n + N = 2450$, choose the 80% observation data as the training set with the sample size $n = 1960$ to learn up and down trends for stock prices and choose the remaining 20% observation data as the test set with the sample size $N = 490$ to predict up and down trends. In this paper we apply the R function getSymbols from the Yahoo Finance port to obtain opening price (O_t), highest price (H_t), lowest price (L_t), closing price (C_t), volume (V_t) and adjusted price (A_t) for Google corporation and then adopt the R package TTR to calculate the 19 technical indicators: WMA, DEMA, ADX, MACD, CCI, Mo, RSI, ATR, CLV, CMF, CMO, EMV, MFI, ROC, VHF, SAR, TRIX, WPR, SNR. In this paper we take Y_t as

Table 4 Two-class confusion matrix

	True class 1($Y_t = 1$)	True class 2 ($Y_t = 0$)
Predicted class 1($\widehat{Y}_t = 1$)	TP	FP
Predicted class 2($\widehat{Y}_t = 0$)	FN	TN

TP True positive, *FP* False positive, *TN* True negative, *FN* False negative

the response variable and 19 technical indicators as the predictor vector to construct the aforementioned five penalized logistic regressions for predicting up and down trends for Google's stock prices. Table 5 lists five summary statistics to the 19 technical indicators and variance inflation factors (VIF) based on the training set $\{x_t, y_t\}_{t=1}^{n=1960}$, where summary statistics show the characteristics of the data, and VIF shows the collinearity relations among 19 technical indicators.

Two indicators WMA_t and $DEMA_t$ represent moving averages of stock prices and mainly show the fluctuation range and dispersion degree of stock prices. From Table 5, we observe that minimum, maximum, median, mean and standard deviation of WMA_t, $DEMA_t$ and SAR_t are larger than those of the other indicators. The mean value of ADX_t indicates that the average degree of trend change of Google stock is 40.1045. $MACD_t$, CCI_t, ATR_t, CLV_t, CMF_t, ROC_t, VHF_t, $TRIX_t$, WPR_t and SNR_t have smaller range, mean and standard deviation. The mean value of momentum line MO_t at 1.9375 reflects the overall upward trend of Google stock price. The mean value of RSI_t is 54.1628, and the maximum value is 98.7890 that is greater than 80 and corresponds to the selling period, whereas the minimum value is 5.5085 less than 10 and corresponds to the buying period. Through the analysis for median and mean to 19 indicators, we found that they are evenly distributed. However, indicators have different degrees of variation, and the values of some indicators differ greatly. Therefore, in order to eliminate the influence of scale variations, we standardize the data before modeling. In order to check whether collinearity exists among 19 indicators, we introduce VIF to check. It can be observed from Table 5 that the VIF for WMA_t, $DEMA_t$ and SAR_t are far greater than 10, and the VIF for MO_t, RSI_t, CMO_t, ROC_t and WPR_t are also greater than 10. This indicates that there exists collinearity among 19 indicators. Thus, it is statistically significant to introduce the penalty functions for logistic regression to reduce collinearity and avoid overfitting.

5.2 Tuning parameter selection

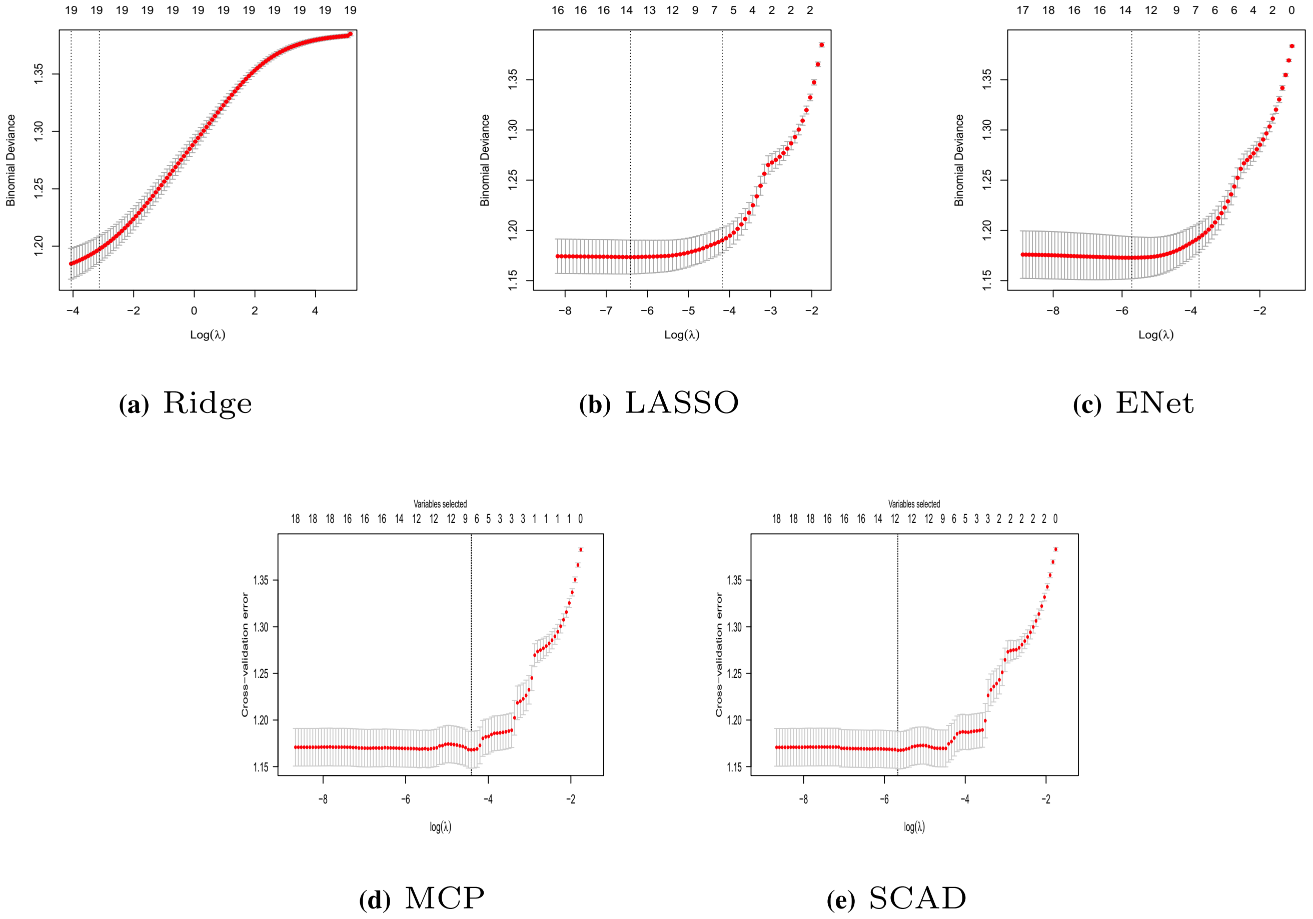
For ridge or LASSO or elastic net penalty, variable selection is determined by the tuning parameter λ. In order to select an appropriate λ, we apply a tenfold cross-validation method to calculate the full solution path to model parameters, select a specific solution path from the full solution path and take the binomial deviation as the risk measure. Then we get the mean cross-validation error curve and the one standard deviation band, see Fig. 1. The parameter estimators for MCP logistic regression and SCAD penalized logistic regression depend on the tuning parameter λ and the regularization parameter γ.

In this section we combine binomial deviation with the tenfold cross-validation method to choose an appropriate tuning parameter λ. Figure 1a, b, c, respectively, represents the binomial deviance curves for ridge, LASSO and elastic net that are drawn by the R function cv.glmnet, whereas Fig. 1d, e, respectively, represents the cross-validation error curves for SCAD and MCP that are drawn by the R function plot.cv.ncvreg. For Fig. 1, the numbers above each graph indicate the selected variable numbers. The left vertical line corresponds to $log(\lambda)$ when the minimum mean square error occurs, the right vertical line represents the corresponding $log(\lambda)$ when 1 times standard error occurs, and $log(\lambda)$ between the two vertical lines indicates that their errors are within a minimum standard error range (i.e., the "one-standard-error" rule). We often use the rule to select the relatively optimum model. From Fig. 1 we observe that the range of "one-standard-error" for ridge, LASSO and elastic net is $0.0173 - -0.0401$, $0.0020 - -0.0154$ and $0.0033 - -0.0213$, respectively. However, for MCP and SCAD, there is only one vertical line and corresponds to the $log(\lambda)$ when the average minimum error occurs, see Fig. 1d, e. We evaluate the prediction performance at each λ and γ value, select the relatively optimum model corresponding to $\lambda = 0.0121$ and $\gamma = 5$ for MCP or $\lambda = 0.0035$ and $\gamma = 10$ for SCAD and obtain the final five penalized regressions. We compare the five penalized regressions with logistic regression and found that ridge logistic regression preserves 19 variables without removing one variable, which is similar to logistic regression, whereas the other four penalized logistic regressions choose different variables, more details see Table 6.

For the five penalized logistic regressions, we calculate their VIF values, see Table 7. From Table 5, we found that the VIF of WMA_t, $DEMA_t$ and SAR_t are 58264.2178, 57089.3227 and 289.9360, respectively, whereas the VIFs of MO_t, RSI_t, CMO_t, ROC_t and WPR_t are greater than 10, which indicates that the strong multicollinearity relations among these indicators exist. From Table 7, we observe that the VIFs of the remaining indicators after the LASSO penalty are all less than 10, after the elastic net, MCP and SCAD

Table 5 Summary statistics and VIF

Indicators	Min	Max	Median	Mean	SD	(VIF)
$X_{(t,1)}$	218.8191	1033.3873	519.5333	512.2369	216.9976	58264.2178
$X_{(t,2)}$	215.8488	1039.1186	517.3452	512.7533	217.4342	57089.3227
$X_{(t,3)}$	10.4160	85.4022	37.9686	40.1045	14.1180	1.2691
$X_{(t,4)}$	−4.4066	5.6060	0.3523	0.4140	1.5782	2.7680
$X_{(t,5)}$	−5.0000	5.0000	0.8121	0.3048	2.8384	9.3459
$X_{(t,6)}$	−86.5400	142.8000	1.8690	1.9375	16.8550	16.5401
$X_{(t,7)}$	5.5085	98.7890	54.7101	54.1628	20.5363	20.6991
$X_{(t,8)}$	2.6340	31.2840	7.6639	9.0489	4.4469	1.8849
$X_{(t,9)}$	−1.0000	1.0000	0.0780	0.0445	0.5999	2.4134
$X_{(t,10)}$	−0.9697	0.7999	0.0381	0.0397	0.2842	2.5886
$X_{(t,11)}$	−100.0000	100.0000	10.9761	8.3479	56.9043	14.4237
$X_{(t,12)}$	−168.6625	57.4554	0.0038	−0.0362	4.0696	1.0212
$X_{(t,13)}$	0.0000	100.0000	53.8371	52.5334	26.6619	4.5240
$X_{(t,14)}$	−0.1384	0.2385	0.0042	0.0034	0.0340	11.4282
$X_{(t,15)}$	0.1232	0.9994	0.5736	0.5849	0.1898	1.2715
$X_{(t,16)}$	216.0054	998.7722	506.7088	509.2556	215.7863	289.9360
$X_{(t,17)}$	−1.4159	2.5168	0.0721	0.0691	0.4112	8.1197
$X_{(t,18)}$	0.0000	1.0000	0.4061	0.4475	0.3113	14.0201
$X_{(t,19)}$	0.0000	4.9967	1.1242	1.3006	0.9341	1.5045



(a) Ridge

(b) LASSO

(c) ENet

(d) MCP

(e) SCAD

Fig. 1 The relationships between binomial deviance/cross-validation error and log(λ)

Table 6 Parameter estimators for logistic regression and five penalized logistic regressions

Coefficient	LR	Ridge	LASSO	ENet	MCP	SCAD
β_0	0.4918	−0.1008	−0.1049	−0.1050	−0.1137	−0.1520
β_1	0.2401	0.0159			0.0311	
β_2	−0.2343	0.0087				
β_3	0.0016	0.0031				
β_4	−0.0192	−0.0383				
β_5	−0.1959	−0.3103	−0.5788	−0.5451	−0.5068	−0.4751
β_6	0.0373	0.0869				
β_7	−0.0313	−0.1191		−0.1234	−1.0464	−1.0114
β_8	0.0214	0.0485	0.0260	0.050	0.0015	0.0956
β_9	−0.2859	−0.1568	−0.1622	−0.1761		−0.0992
β_{10}	0.3466	0.1977	0.0917	0.1443		0.1333
β_{11}	0.0208	0.4683	0.8422	0.8287	1.4888	1.6237
β_{12}	0.0507	0.0258				0.0488
β_{13}	0.0145	0.3582	0.3496	0.3981	0.4470	0.3876
β_{14}	−9.7227	0.1061				−0.2134
β_{15}	0.6910	0.0715	0.0101	0.0361	0.0122	0.0975
β_{16}	−0.0057	0.0033				
β_{17}	1.3665	0.1115		0.0368	0.4117	0.4375
β_{18}	−0.9247	0.1081				
β_{19}	−0.0983	−0.0250				−0.0918

LR represents logistic regression

Table 7 VIF for the remaining variables

Variables	VIF ($LASSO$)	VIF (ENet)	VIF (MCP)	VIF (SCAD)
$X_{(t,1)}$			1.7888	
$X_{(t,5)}$	2.5307	4.4068	4.3680	4.6557
$X_{(t,7)}$		14.1372	11.7272	15.1485
$X_{(t,8)}$	1.0041	1.0078	1.7588	1.0141
$X_{(t,9)}$	1.3427	1.6880		1.7260
$X_{(t,10)}$	2.2507	2.3057		2.3101
$X_{(t,11)}$	6.1593	7.9904	7.3606	10.8813
$X_{(t,12)}$				1.0134
$X_{(t,13)}$	4.3145	4.4449	4.2970	4.4678
$X_{(t,14)}$				5.2576
$X_{(t,15)}$	1.0081	1.0082	1.0106	1.2629
$X_{(t,17)}$		4.1359	3.4627	5.3538
$X_{(t,19)}$				1.3270

penalty, only the VIFs of RSI_t are greater than 10, which are 14.1372, 11.7272 and 15.1485, respectively. Therefore, penalized logistic regressions can greatly weaken or eliminate collinearity relations among technical indicators.

5.3 The prediction performance

We take advantage of the training set $\{x_t, y_t\}_{t=1}^{n=1960}$ to learn up and down trends for Google's stock price and apply the testing set $\{x_t, y_t\}_{t=1961}^{2450}$, and the ROC curve to evaluate the prediction performance. According to the predicted class from the training set and the actual class from the testing set, we establish the following two-class confusion matrix, see Table 8.

From Table 8 we calculate accuracy, sensitivity and specificity for logistic regression as follows:

Table 8 Two-class confusion matrix

	Actual 1($Y_t = 1$)	Actual 2 ($Y_t = 0$)
Predicted 1($\widehat{Y}_t = 1$)	191	84
Predicted 2($\widehat{Y}_t = 0$)	51	164

Table 9 The prediction performances for the six methods

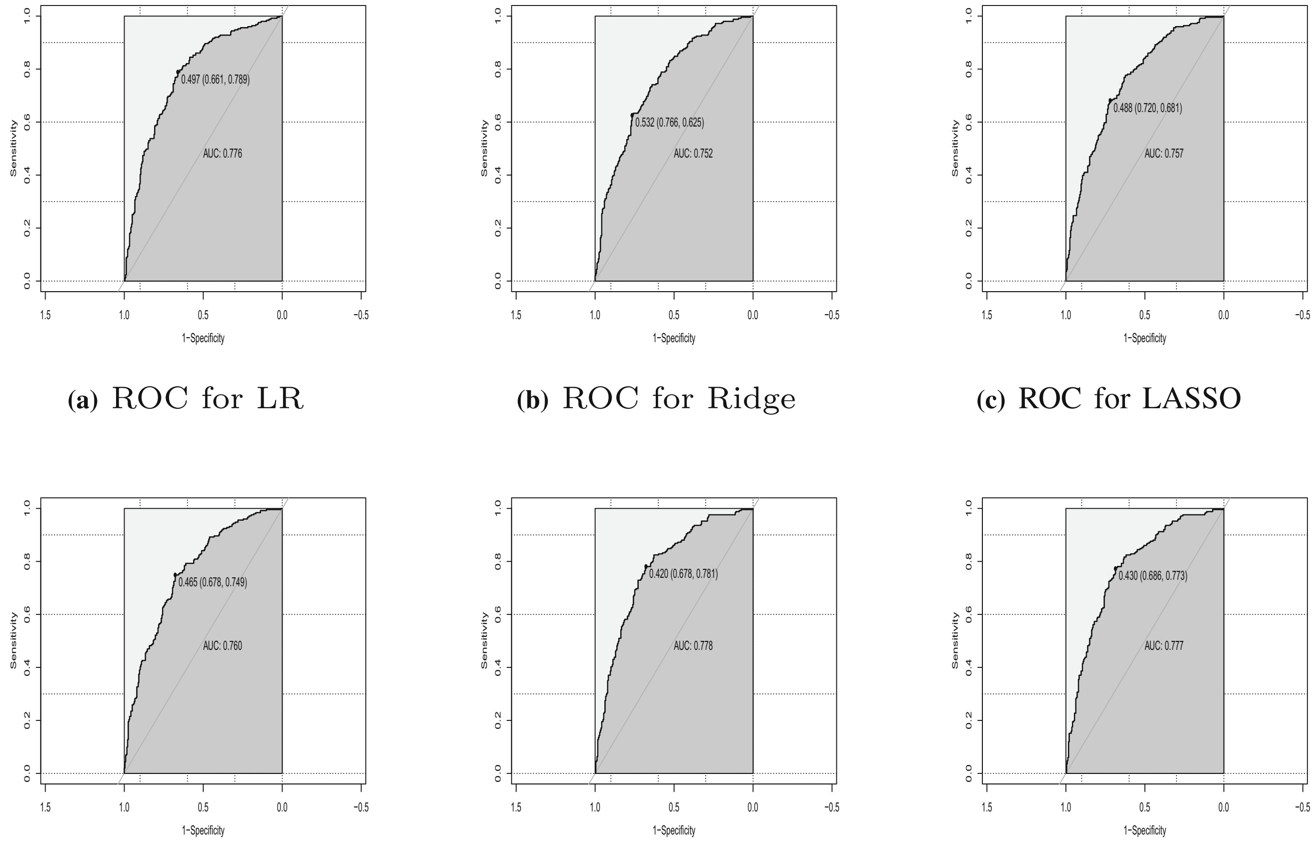
	LR	Ridge	LASSO	ENet	MCP	SCAD
Sensitivity	0.789	0.625	0.681	0.749	0.781	0.773
Specificity	0.661	0.766	0.720	0.678	0.678	0.686
Accuracy	0.724	0.694	0.705	0.712	0.732	0.731

$$\text{Accuracy} = \frac{191 + 164}{191 + 164 + 84 + 51} \approx 0.724,$$
$$\text{Sensitivity} = \frac{191}{191 + 51} \approx 0.789,$$
$$\text{Specificity} = \frac{164}{164 + 84} \approx 0.661.$$

Similarly, we calculate accuracy, sensitivity and specificity for the five penalized logistic regressions. Their specific values are listed in Table 9.

From Table 9 we observe the following facts: (1) For elastic net and LASSO, accuracy is higher than that of ridge, but is lower than that of logistic regression; (2) accuracy for MCP is higher than that of SCAD, whereas accuracy for SCAD is higher than that of elastic net and logistic regression. However, accuracy is the simplest index to evaluate the prediction, and it cannot fully reflect the corresponding loss of two kinds of errors. Therefore, in the following we first compute sensitivity and specificity corresponding to different thresholds for the six methods and then apply them to draw the ROC curve to evaluate accuracy, see Fig. 2.

In Fig. 2, the AUC corresponding to logistic regression, ridge, LASSO, elastic net, MCP and SCAD is 0.776, 0.752, 0.757, 0.760,0.778 and 0.777, respectively. Combined with accuracy listed in Table 9, it can be concluded that among the six methods, the MCP logistic regression with technical



(a) ROC for LR **(b)** ROC for Ridge **(c)** ROC for LASSO

(d) ROC for ENet **(e)** ROC for MCP **(f)** ROC for SCAD

Fig. 2 The ROC curves for the six models

Table 10 Sensitivity, specificity, accuracy and AUC for MCP, SVM and ANN

	MCP	SVM	ANN
Sensitivity	0.781	0.705	0.725
Specificity	0.678	0.653	0.732
Accuracy	0.732	0.686	0.729
AUC	0.778	0.679	0.759

indicators performs the best in terms of in terms of accuracy. In order to further explain the superiority to the MCP logistic regression in predicting stock prices trends movement, we compare the prediction performances for the MCP logistic regression with those for SVM and ANN, see Table 10.

From Table 10, we can observe that among the aforementioned three methods, MCP performs the best in terms of sensitivity, accuracy and AUC. The reason that SVM performs the worse may be that Gaussian kernel function is a typical local kernel function, and it only affects the data points in a small area near the test point and has strong learning ability and weak generalization performance. In addition, ANN is unstable, so we choose the average of the 10 predicted results as the final values, and they are worse than MCP. Obviously, the MCP logistic performs best in predicting the trend of stock price ups and downs. Therefore, we recommend the MCP logistic regressions to predict the stock price trend movements.

6 Discussion

Methodologically, we introduce the five penalty functions to logistic regression with 19 technical indicators and propose the five penalized logistic regressions to predict up and down trends for Google's stock prices. These prediction methods not only can provide classification probability estimation and class index information, but also improve the prediction accuracy by shrinking regression coefficients and avoiding multicollinearity and overfitting. Computationally, we combine the iteration weighted least squares, the coordinate descent algorithm and the tenfold cross-validation method for the five penalized logistic regressions to obtain their parameter estimations and probability estimations. According to the VIF analysis in Table 5, we found that there exists collinearity among the different technical indicators. Thus, it is statistically significant to introduce the different penalty functions to reduce collinearity relations in logistic regression with 19 technical indicators. Therefore, we propose the five efficient penalized logistic regressions to predict stock price trend movement. Wen et al. (2019) and Khan et al. (2020) predicted Google stock trend movements, whose accuracy is 0.636 and 0.641, respectively. From Table 9 we observe that the prediction accuracies of the five penalized logistic regressions are higher than 0.693. In particular, the prediction accuracies of MCP and SCAD are 0.732 and 0.731, respectively. The AUCs of MCP and SCAD are 0.778 and 0.777, respectively. Obviously, MCP and SCAD penalized logistic regressions outperform logistic regression in terms of the prediction performance. Furthermore, compared MCP and SCAD with SVM and ANN, we found that the proposed MCP and SCAD penalized logistic regression performs better than SVM and ANN. Therefore, in this paper we provide the new methods to predict stock market trends movement. Moreover, the proposed methods help investors to better understand the internal mechanism of stock market trends movement.

7 Conclusion

Based on Murphy's technical analysis method, we combine technical indicators with five penalized logistic regressions and propose the five penalized logistic regressions to predict the up and down trends of Google's stock price. The prediction results show that the MCP logistic regression with technical indicators is superior to logistic regression, the other four penalized logistic regressions, SVM and ANN. Therefore, in this paper we combine technical indicators with MCP logistic regression and provide the new effective prediction method to further improve the prediction performance to stock returns. For other stock price trends prediction problems, we can also apply statistical charts, data analysis, empirical knowledge and the penalized method to extract some important technical indicators that may affect stock price trends movement, establish some penalized logistic regressions with different technical indicators to predict up and down trends for stock prices and apply the two-class confusion matrixes and ROC curves to assess their prediction performances.

Author Contributions XH provided the basic idea and improved the writing to the manuscript. HJ collected data, provided the figures and tables, and finished the basic writing. HJ improved the program.

Funding This research was supported by the Fifth Batch of Excellent Talent Support Program of Chongqing Colleges and University (68021900601), the Natural Science Foundation of CQ CSTC (2018jcyjA2073), Science and Technology Research Program of Chongqing Education Commission (KJZD-M202100801), the Program for the Chongqing Statistics Postgraduate Supervisor Team (yds183002), Chongqing Social Science Plan Project (2019WT59,2020YBTJ102), Open Project from Chongqing Key Laboratory of Social Economy and Applied Statistics (KFJJ2018066) and Mathematic and Statistics Team from Chongqing Technology and Business University (ZDPTTD201906).

Data availability The datasets analyzed during the current study are available in the Yahoo Finance, uk.finance.yahoo.com.

Declarations

Conflict of interests The author declares that they have no relevant financial or non-financial interests to disclose.

Ethical approval This article does not contain any studies with human participants or animals performed by the author.

References

Breheny P, Huang J (2011) Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. Annals of Applied Statistics 5(1):232–253

Cavalcante RC, Brasileiro RC, Souza VLF, Nobrega JP, Oliveira ALI (2016) Computational intelligence and financial markets: a survey and future directions. Expert Systems with Applications 55(15):194–211

Elliott G, Granger C, Timmermann A (2013) Handbook of economic forecasting. North Holland Elsevier

Hu XM, Jiang HF (2021) Logistic regression model with technical indicators predicts ups and downs for google stock prices. System Science and Mathematics 41(3):1–22

Hu XM, Liu F (2020) Estimation theory and model recognition for high-dimensional statistical models. Higher Education Press, Beijing

Joshi K, Bharathi HN, Rao J (2016) Stock trend prediction using news sentiment analysis. International Journal of Computer Science and Information Technology 8(3):67–76

Khan W, Malik U, Ghazanfar MA, Azam MA, Alyoubi K, Alfakeeh A (2020) Predicting stock market trends using machine learning algorithms via public sentiment and political situation analysis. Soft Computing 24(15):11019–11043

Li JH, Bu H, Wu JJ (2017) Sentiment-aware stock market prediction: a deep learning method. International Conference on Service Systems and Service Management 202:1–6

Li S, Ning K, Zhang T (2021)?Sentiment-aware jump forecasting. Knowledge-Based Systems 228: 107292

Malandri L , Xing F Z , Orsenigo C , Vercellis C (2018) Public moodC-driven asset allocation: the importance of financial sentiment in portfolio management. Cognitive Computation 10: 1167C1176

Murphy J J (1999) Technical analysis of the financial markets. New York Prentice Hall Press

Nabipour M, Nayyeri P, Jabani H, Shahab S, Mosavi A (2020) Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis on the Tehran stock exchange. IEEE Access 99(8):150199–150212

Nair B, Sai SG, Naveen AN, Lakshmi A, Venkatesh GS, Mohandas V (2011) A ga-artificial neural network hybrid system for financial time series forecasting. Information Technology and Mobile Communication 147(2):499–506

Picasso A, Merello S, Ma YK, Oneto L, Cambria E (2019) Technical analysis and sentiment embeddings for market trend prediction. Expert Systems with Applications 135:60–70

Shen JY, Shafiq MO (2020) Short-term stock market price trend prediction using a comprehensive deep learning system. Journal Of Big Data 7(1):66–98

Wang L, Zhu J (2010) Financial market forecasting using a two-step kernel learning method for the support vector regression. Annals of Operations Research 174(2):103–120

Wen M, Li P, Zhang LF, Chen Y (2019) Stock market trend prediction using high-order information of time series. IEEE Access 7:28299–28308

Xing F Z , Cambria E , Malandri L , Vercellis C (2018) Discovering bayesian market views for intelligent asset allocation. Machine Learning and Knowledge Discovery in Data bases 9(2): 120C135

Xing FZ, Cambria E, Welsch RE (2018) Intelligent asset allocation via market sentiment views. IEEE Computational Intelligence Magazine 13(4):25–34

Xing F Z, Cambria E, Zhang Y (2019) Sentiment-aware volatility forecasting. Knowledge-Based Systems 176(JUL.15):68-76

Zhang J, Cui SC, Xu Y (2018) A novel data-driven stock price trend prediction system. Expert Systems with Applications 97(1):60–69

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.