



# US Options Daily Analytics Guide

version 1.1 (Feb 2023)



**algoseek | the market data company**

We provide research market data for machine learning and quantitative trading



## CONTACT US

We are here to help you do great things with our market and reference data. For questions, feedback, and other concerns, you may reach our team of experts using the following contact information:

### **algoseek customer support**

✉ [support@algoseek.com](mailto:support@algoseek.com)

☎ (+1) 646 583 1832

### **algoseek sales**

✉ [sales@algoseek.com](mailto:sales@algoseek.com)

☎ (+1) 646 583 1832



## TABLE OF CONTENTS

<b>INTRODUCTION</b>	<b>4</b>
<b>Pricing Model</b>	<b>4</b>
American Options	4
European Options	4
Implied Volatility	4
Interest Rates	5
Dividend Projections	5
Borrow Rates	6
Corporate Actions and Non-Standard Deliverables	6
<b>DATA ORGANIZATION AND FILE FORMAT</b>	<b>6</b>



## INTRODUCTION

Algoseek offers an enhanced US Options Analytics dataset that includes implied volatility and Greeks for the last minute mid-price of each traded contract. All calculations use our published open-source methodology. This dataset is based on algoseek's Equity Market and Reference datasets, Options and Futures Market datasets, and IBA LIBOR data.

## Pricing Model

### Black-Scholes-Merton Model

The Black-Scholes model (BS) is the most important concept in modern financial theory for pricing option contracts. This mathematical equation estimates the theoretical value of an option contract, taking into account the impact of time and other risk factors (underlying price, the risk-free rate, and the volatility). The model can be extended to value options on instruments paying dividends, known as the Black-Scholes-Merton model (BSM). Equations from both models can be used for pricing European options. Since the American options can be exercised before the expiration, BS and BSM equations become variational inequalities. In this case, different approaches can be used to find numerical solutions (finite-difference model, binomial model).

### Option Greeks

Greeks are derivatives that measure the sensitivity of an option's price relative to various factors, such as underlying price (Delta, Gamma), time to expiration (Theta), volatility (Vega), and interest rate (Rho). Delta is the most important of the option Greeks. It measures the sensitivity of an option's price (premium) to changes in the underlying price. It can be treated as the speed of the option's price change. In the meantime, Delta changes with the underlying price. Another greek, Gamma, measures these changes and can be treated as the acceleration of the option's price change. Options generally lose their value with the passing of time. This process is known as time decay. Theta measures the speed of time decay – how much option premium will decrease in one day. Vega measures the sensitivity of an option to volatility. Rho measures an option's sensitivity to changes in interest rate.

### American Options

For American Options a Black Scholes finite-difference pricing model with implicit time stepping is used. The calculation uses the QuantLib's pricing engine `FdBlackScholesVanillaEngine` described in the Python documentation at [https://quantlib-python-docs.readthedocs.io/en/latest/pricing\\_engines.html](https://quantlib-python-docs.readthedocs.io/en/latest/pricing_engines.html) with C++ source <https://github.com/lballabio/QuantLib/blob/master/ql/pricingengines/vanilla/fdblackscholesvanillaengine.cpp>. algoseek forecasted discrete dividends for stocks and ETFs as of the



calculation date are used within the model. More details on the dividend projections are provided below.

For numerical calculations, algoseek uses a uniform rectangular grid with 300 time steps and 900 price steps. The values of Delta, Gamma, and Theta are taken directly from QuantLib library and are calculated from the underlying grid. The rest of the Greeks (Rho and Vega) are calculated numerically using central finite differences. Rho is estimated by bumping the risk-free rate to maturity by ten basis points and Vega bumps volatility by hundred basis points.

## European Options

European option Greeks are calculated using the closed-form solution to the Black-Scholes model with dividend yield (Haug, 2007). algoseek estimates the dividend yields for index options by relying on the put-call parity relation that holds for European options. This process is described in more detail below.

## Implied Volatility

algoseek attempts to calculate implied volatility for the last quote mid-price. The implied volatility is capped at 900%. If the bid is less than or equal to a penny or if it is less than the intrinsic value of the option, we do not compute implied volatility directly but instead, attempt to estimate it from nearby strikes.

If we are able to compute the implied volatility of nearby strikes of the same maturity we use linear interpolation to estimate the volatility in the Greeks calculations. For options that do not have nearby strikes to interpolate from, we try to get implied volatility from the put-call pair. And if we can't do this, we use simple constant extrapolation from the last successful calculation at the high and low strikes.

We include the ImpliedVolConvergence field for each implied volatility to indicate if the value was the result of root convergence, capping, interpolation, or extrapolation.

## Interest Rates

algoseek constructs interest rate yield curves based on IBA LIBOR and CME Eurodollar futures. IBA LIBOR rates for up to one month period are used, followed by Eurodollar futures extending out to two years. The yield curve is generated for each trade date. Interpolation to the option maturity is then used to determine a risk-free rate. From 20220701 algoseek switched to SOFR and SOFR futures. Interest rate yield curves are built based on [CME methodology](#). There are a couple of differences comparing algoseek methodology with CME: 1) Take into account just settlements for SOFR futures as inputs to optimization function and ignore intraday bid/ask/trade events; 2) Use 10 consecutive quarterly contracts for 3-month SOFR futures; 3) Extend term structure up to 24-month tenor.



## Dividend Projections

Valuing options requires a forecast of dividends that the underlying stock, ETF, or index constituents will pay. For stocks and ETFs we use a simple model to project the expected dividend amounts and dates based on historical and announced payments. For index options, we imply the dividend yield from option quotes.

### Single Stocks and ETFs Dividend Projections

A simple “martingale” method of projection is used for single stocks and ETFs. We assume that future payments will be the same amount as the last regular dividend payment and at the historical frequency for the stock.

The dividend projections are handled in three steps:

Step 1: Determine the frequency with that the stock or ETF has paid dividends in the past: monthly, quarterly, semi-annually, annually, or does not pay dividends.

Step 2: Calculate the expected payment dates in the future given the frequency of payment and the date of the last historical or announced ex-date.

Step 3: Assume the amount of the dividend on all future dates matches the current announcement or last paid regular dividend.

### Index Dividend Projections

For index options, we use an estimate of the dividend yield that would be gained if you owned the constituents in the same ratio as the index. Index dividend projections are calculated daily using closing prices.

The dividend yield is implied by assuming put-call parity holds for a specific strike and maturity. Using the put-call parity equation it is possible to back out an implied dividend yield for European options.

$$q_T = -\frac{1}{T} \ln \frac{c - p + Ke^{-r_T T}}{S},$$

where  $q_T$  is the dividend yield to  $T$ ,  $c$  is the call price,  $p$  is the put price,  $K$  is the strike of the option,  $T$  is the time to maturity for the option,  $r_T$  is the risk-free rate to maturity of the option, and  $S$  is the underlying price.

For each maturity, the average of the five closest at-the-money strikes is used to make a point estimate. The point estimates are then used to fit a linear (in time) curve across the median dividend yields at each maturity.



## Borrow Rates

For non-index options borrow rates are implied using prior-day closing option prices. A borrow rate may be inferred by minimizing the difference of implied volatilities for at-the-money puts and calls at each maturity using the sum of squares of difference as an objective function. There is an alternative way to imply borrow rates that allows us to avoid the minimization problem (<https://dx.doi.org/10.2139/ssrn.2851560>). We consider the five closest to at-the-money strikes for each maturity greater than 28 days in the future and find the borrow rate using an approximated formula that links together the borrow rate and implied volatility spread. For each maturity, the median of the five closest at-the-money strikes is used to make a point estimate. The point estimates are then used to fit a median (in time) curve across the median borrow rates at each maturity.

## Corporate Actions and Non-Standard Deliverables

When a company has a stock split, merger, spin-off, special dividend, or other corporate action, option contracts on the stock may be modified. The Options Clearing Corporation (OCC) publishes the modifications daily on its website (<https://infomemo.theocc.com/infomemo/search>). Because these adjusted options may deliver multiple stocks and cash in different quantities, they require special handling in options models.

Currently, algoseek does not handle non-standard deliverables and will instead set the relevant fields to zero. Also, algoseek doesn't cover VIXW options, options on FX, and options on underlying securities that are traded OTC.

## DATA ORGANIZATION AND FILE FORMAT

algoseek provides Options Analytics data in plain-text CSV files. The first row of the CSV file is a fixed header, and then rows of data corresponding to individual events (see Table 1). By default, data is organized into one file per symbol per trading day. For example, all data for ticker AAPL on Mar 3, 2020, are stored in one CSV file.

**Table 1: CSV File Fields Schema**

Field	Type (Format)	Missing	Description
TradeDate	string (yyyymmdd)	Never	Trading date in yyyymmdd format
Ticker	string	Never	Symbol name
CallPut	string	Never	Option contract type: "C" for Call or "P" for Put



OptionStyle	string	Never	Option exercise style: "A" for American or "E" for European
Strike	decimal	Never	A fixed price for buying or selling an option contract
Expiration	string (yyyymmdd)	Never	The expiration date of option contract in yyyymmdd format
YearsToMaturity	decimal	Never	The amount of time in years until the contract expires
DaysToMaturity	integer	Never	The number of days until the contract expires
UnderLastMidPrice	decimal	Never	Underlying NBBO mid-price at the last minute traded
UnderLastMidTime	decimal	Never	The minute bar for UnderLastMidPrice
LastBidPrice	decimal	Blank	The option NBBO bid price at the last minute traded
LastBidTime	string	Blank	Timestamp for LastBidPrice
LastMidPrice	decimal	Blank	The option NBBO mid-price at the last minute traded
LastAskPrice	decimal	Blank	The option NBBO ask price at the last minute traded
LastAskTime	string	Blank	Timestamp for LastAskPrice
MidImpliedVol	decimal	Never	The implied volatility as per LastMidPrice
MidTheoPrice	decimal	Never	The theoretical price calculated for LastMidPrice
MidDelta	decimal	Never	The change in option price with respect to the underlying price
MidGamma	decimal	Never	The change in option Delta with respect to the underlying price
MidTheta	decimal	Never	The change in option price with respect to time
MidVega	decimal	Never	The change in option price with respect to implied volatility
MidRho	decimal	Never	The change in option price with respect to interest risk-free rate
ImpliedVolConvergence	string	Never	Convergence code for implied volatility

All possible codes for the ImpliedVolConvergence field with their descriptions can be found in the table below.

**Table 2: ImpliedVolConvergence Codes**

Code	Description
Converged	Implied volatility was calculated for an option contract within the pricing model
IntrVal_LinInterp	Option bid price is less than the intrinsic value. Implied volatility was estimated using linear interpolation
SmallBid_LinInterp	Option bid price is less or equal to \$0.01. Implied volatility was estimated using linear interpolation
Failed_LinInterp	Implied volatility couldn't be calculated for an option contract. Implied volatility was estimated using linear interpolation
IntrVal_PutCallPair	Option bid price is less than the intrinsic value. Implied volatility was estimated from put-call parity
SmallBid_PutCallPair	Option bid price is less or equal to \$0.01. Implied volatility was estimated from put-call parity
Failed_PutCallPair	Implied volatility couldn't be calculated for an option contract. Implied volatility was estimated from put-call parity
IntrVal_FlatExtrapol	Option bid price is less than the intrinsic value. Implied volatility was estimated using constant extrapolation
SmallBid_FlatExtrapol	Option bid price is less or equal to \$0.01. Implied volatility was estimated using constant extrapolation
Failed_FlatExtrapol	Implied volatility couldn't be calculated for an option contract. Implied volatility was estimated using constant extrapolation
Failed	Implied volatility couldn't be calculated within the pricing model or estimated for an option contract