



Study guide for Data Analyst Level 1 certification

Skip Ahead

[Align Experimentation to Your Business](#)

[Align Experiments with Business Goals](#)

[Building a Goal Tree](#)

[Creating a Testing Roadmap](#)

[Creating an Experimentation Roadmap](#)

[Understanding Primary and Secondary Metrics](#)

[Experiment Duration and Sample Size](#)

[Optimization statistics](#)

[What is Statistical Significance?](#)

[What are Statistics, Anyway?](#)

[What is the Stats Engine?](#)

[Understanding the Results Page](#)

[Interpreting the Results page](#)

[Filter your results and segmentation](#)

[Share your results with others](#)

[How do you share results?](#)

[What are the best tools for sharing results?](#)

[Take action on winning variations](#)

[Managing winning variations](#)

[How do I implement my winners?](#)

[Take action on losing variations](#)

[What do I do with losing variations?](#)

[Taking action on inconclusive results](#)

[What is an inconclusive result and what can I do with it?](#)

[How long to wait for conclusive results](#)

This article will help you prepare for your Optimizely Data Analyst Level 1 exam where you can earn the Data Geek Level 1 certification. Good luck!

Align Experimentation to Your Business

Align Experiments with Business Goals

- Business goals are as important as having a destination when on a road trip.
- Experiments must align with metrics your business most cares about.
- With each experiment, you should be able to express and predict how changes will:
 - Affect user behavior
 - Directly benefit a business goal
- Using an experiment to settle an argument about which design is better, or to answer the question “I wonder what happens if I do xyz...” isn’t the best priority: **always make your business goals a priority.**
- Consider a hierarchy of goals to help you iterate your experiment in order to influence your highest goal of the company:
 - Company goal: increase total revenue
 - Business Unit goal: increase revenue per visit
 - Optimization goal
 - Experiment goal: more granular (like clicks to add to cart button) a concrete action
- Doing this creates your **goal tree**, where you can see where your experiment goals can work up and up towards the ultimate company goal.

Building a Goal Tree

- The purpose of creating goal tree is to organize the metrics that feed into the company goal. It can also help you decide which goal to pursue first.
 - It’s the foundation of your ideation strategy.

To go deeper on this subject, check out our article on [primary and secondary metrics and monitoring goals](#).

Creating a Testing Roadmap

Creating an Experimentation Roadmap

- There are four questions to ask yourself:
 - What are we experimenting on?
 - When are we experimenting?
 - Who is involved in experimentation? (Establish timeline for reviews, etc., to ensure people are available when you need them.)
 - How will our experiments align with company-wide strategic objectives? (This is a good question for executive-level folks. Experiments must align with KPIs to make sure they're relevant, and to increase likelihood that they will yield a real impact.)

To go deeper on this subject, check out our article on [creating an experimentation roadmap](#).

Understanding Primary and Secondary Metrics

- Consider primary and secondary metrics for your experiments: the primary metric is your goal, while secondary metrics are monitoring goals.
 - Primary metric: the yardstick by which you can tell if your experiment was a success.
 - Secondary: supporting events that provide more insight and connection to overall business goal; these also give visibility across different steps along the funnel.
 - Experimenting further down-funnel can make statistical significance harder for experiments to achieve at all.
- Your business goal is not necessarily going to match your primary experiment goal.

Experiment Duration and Sample Size

- You can't properly plan or roadmap until you know how long your experiments will take to reach statistical significance.
 - In order to determine how long an experiment will run, you'll need an estimate of your sample size (the number of people who will be exposed to experiment)
 - Optimizely's stats engine uses a process called **sequential testing** that collects evidence as your test runs to flag when your experiment reaches stat sig, so you can see winners and losers as quickly and accurately as possible.

- The sample size calculator can help you determine the projected stopping point for an experiment.
- If you don't have an analytics platform to tell you your baseline conversion rate, you can use Optimizely and just run an experiment without a variation for a predetermined amount of time.
- **Minimum detectable effect:** the percentage lift you want to detect. This helps you clarify the likely relationship between impact and effort.

Optimization statistics

What is Statistical Significance?

- Statistical significance is a mathematical demonstration that your results are reliable.
- The higher you set [statistical significance](#), the higher the probability that those results are accurate and not due to random chance.
- There is no such thing as **complete** certainty in statistics.
- The significance level you choose is a reflection of how comfortable you are with risk. The lower the number, the more comfortable you are with uncertainty in your results.
- Optimizely's default statistical significance setting is 90%.
- You need a certain number of data points (at least 100 visitors and 25 conversions on each variation to reach this at Optimizely) in order to generate any reliable measurement of statistical significance.
- With a high significance level and low traffic, stat sig will take much longer to achieve.
- The more visitors you get, the more accurately you can make data-driven decisions that will impact your business goals.

To go deeper on this subject, check out our article on [statistical significance in Optimizely](#).

What are Statistics, Anyway?

- In Optimizely, statistics provide insight into your results and give you confidence in your winning variations.
- The **p-value** is the chance that your results would be the same if there were no difference between the variations. If that chance is less than 5%, the results are statistically significant.
- Never end an experiment too early, even if it reached statistical significance for the first time (even if it happens before reaching required sample size). Doing so could give you bad data.

What is the Stats Engine?

- It's a proprietary system that combines innovative statistical methods to give you trustworthy data faster.
- **Sequential testing**: a technique designed to evaluate experiment data as it is collected. This is different from traditional statistics that assume you will only evaluate your experiment data at one point in time, at a set sample size.
- Sequential testing calculates an average likelihood ratio—the relative likelihood that the variation is different from the baseline—every time a new visitor triggers an event.
- **False discovery rate control** allows for many goals and variations with guaranteed accuracy. If you report a rate of 10% it means that, at most, 10% of winners and losers have no difference between variation and baseline, the same chance of making an incorrect business decision.
- With the Stats Engine, winners and losers are reported with low false discovery rate instead of low false positive rate. As goals and variations are added to the experiment, Optimizely corrects more for false discoveries and becomes more conservative in calling winners/losers.
- This control gives you transparent assessment of the risk you have of making an incorrect decision.
- It also gives you the ability to continuously monitor the data coming in in real time.

For more detailed information on Stats Engine, check out [this video](#) and our article about [how Optimizely's Stats Engine calculates results](#).

Understanding the Results Page

Interpreting the Results page

- Example experiment: image on product detail page
 - Lifestyle imagery vs product imagery
 - Access results by clicking on your experiment and clicking *Results*.
 - Summary has a high-level overview of the experiment: it shows improvement to primary metric for each variation compared to baseline.
 - Your primary metric is always at the top: conversions are also there, calculated by uniques and totals.
 - A unique conversion occurs only the first time a visitor triggers a goal.
 - Total conversions count each time a visitor triggers a goal.
 - Why are they useful?

- Unique conversions help you control for outliers and track conversions such as clicking the Add To Cart button.
- Total conversions help you track repeat behavior: for example, social media sharing or viewing multiple articles on a media site.
- Can we be confident of value of the change?
 - The **confidence interval** measures uncertainty around improvement. It's a range, and the conversion rate for a particular experience lies within it. It starts out wide, but as more data is collected, the interval narrows to show certainty is increasing.
 - Once statistical significance is reached for a variation, the interval is either all above or all below 0.
 - Confidence interval is set at the same level you chose for the project's statistical significance threshold.
 - Your experiment must collect enough data to declare statistical significance. If you don't have enough, Optimizely will give you an approximation of how long that might take.

Filter your results and segmentation

- One example of segmentation might be filtering results based on a certain time period. This is commonly done for holidays, to learn if that holiday affected visitor behavior.
- Filters include: data range, type of user (desktop vs. mobile), gender, location.
 - Some filters are already in the system, while others are custom and can be added by you (filters are called **attributes** in the system).
 - Default filters: Browser, Source (Campaign, direct, referral, search), Campaign, Referrer, Device.
 - Filters are useful if you have two variations that beat your baseline, and you want to compare them to each other.
 - The results page can be shared with stakeholders by sharing the URL. You can reset this link, but if you do the original link will no longer work.

To learn more, read our article about [the experiment results page for Optimizely](#).

Share your results with others

How do you share results?

- Clearly express the purpose of the experiment: what inspired your hypothesis? Engage the company in the story of why the experiment was run in the first place.
- Describe the construction of the experiment: numbers of variations, description of differences between variations, dates of experiment, visitor count, and visitor count per variation.
- Be concrete: include the lift, your interpretation of the data, etc. Quantify given percentage lift with year-over-year projected change. Explain how the winning variation contributes to company's KPIs.
- Consider using a template for hard data, so you can re-use it each time.
- Although more difficult, it can be helpful to share inconclusive or losing results too: these provide value, and segmenting inconclusive results can deliver more insight.
- Include lessons learned from losing experiments.

What are the best tools for sharing results?

- Spreadsheets, newsletter, blog, and roadshows are all commonly used, but you should use whichever channel will be most effective within your organization.

Curious about templates for sharing your results with stakeholders? [Check out this article!](#)

Take action on winning variations

Managing winning variations

- You should implement the winning variation first: you can do this within the experiment itself by allocating 100% traffic to the winning variation, which temporarily sends all visitors to the winning variation.
- Get more winners by expanding a successful hypothesis: can you optimize in the same way elsewhere?
- Leverage the method that made you successful: can you apply the change you made on your product page to your home page?
- Base experiments on solid hypothesis supported by rational analysis of your data.
- Remember, experimentation is for *learning*, not just for a lift (positive results).
- Each time an experiment concludes, that's not an ending. Instead, it's a step forward in your journey of optimization.

How do I implement my winners?

- Loop in developers early, and update them regularly: this will smooth the process of implementing permanent changes, and build a stronger optimization culture in the process.
- To implement changes, your developer will translate Optimizely's jQuery variation code into the framework your site is built on. Alternately, you could potentially just add Optimizely's variation code directly to your website, but this won't be sustainable, especially if you do implement multiple changes from multiple experiments.
- Implementation may require translating client-side script to server-side script.
- Share your results document with your developer to give them a bigger picture, but also to create a quick proof-of-concept before fully dedicating developer resources.
- Only site changes that have been tested and validated with data should be passed to developers for implementation. Integrate your communications about the site build through JIRA or Trello, or whatever communications system you use.
- Also, create a unified record to avoid communication gaps and prevent conflicts that may break the site.

Sample workflow:

1. Create an experiment based on your experiment plan, with your developer's input.
2. Analyze the results of an experiment and make decisions based on your results.
3. If a variation is successful, duplicate the experiment and set traffic allocation to 100% to show all visitors.
4. Share results with stakeholders, including the developer.
5. Developer enters Optimizely experiment and reviews variation code with the project manager.
6. Developer abstracts the code and builds a permanent feature in the production environment; then, verifies that the new feature works correctly.
7. Developer notifies the optimization program manager when the feature goes live.
8. Optimization program manager turns the experiment off and archives it.
9. Implement the same feature in your development environment, if you have one.
10. Integrate your implementation process with the systems your developer uses to track projects to execute the handoff smoothly.

You and your developer should keep lines of communication open, even during experimentation: the more involved the developers are, the faster and easier it will be to implement winning changes to your site.

Read this article to find out more about [implementing wins on your production site](#).

Take action on losing variations

What do I do with losing variations?

- You can still learn a lot from a loss—maybe even more than you can from a winner:
 - You know conclusively what NOT to do.
 - Now that you know what your visitors don't like, consider other related ventures: what else might visitors dislike along the same lines? Or what might they like? What did you learn about your visitors' habits or preferences?
 - Look at your hypothesis: Maybe the solution didn't speak to your problem the way you thought it did.
 - Ask *why?*: Why did visitors respond badly? Was the change too different? How could we have minimized the negative reaction of the change?

[Segmenting your results can also help!](#)

Taking action on inconclusive results

What is an inconclusive result and what can I do with it?

- This happens when Optimizely isn't confident enough in the results to declare a winner or a loser.
- There will always be the temptation to try and stick out an experiment for longer.
- Why does this happen?
 - Sample size is too small: keep the experiment up to get a larger sample.
 - The change is too small: small tweaks often don't move the needle much.
 - The experiment isn't getting enough traffic: try broadening your URL targeting (maybe try targeting multiple pages).
 - You might have experimented with something your visitors just don't care about: either the problem you're trying to solve doesn't actually exist, or the solution you're trying isn't the right one.
- So what can you do with the data you have?
 - Check your secondary and monitoring metrics: did any of them reach significance?
 - Look for trends and patterns. Your confidence interval can help you do this: with no statistical significance, it can give you a range of values at least, which might help you make a confident decision.

- Segment any inconclusive results: break them out and see how they behave differently. You might find conclusive results among your segments.
- Remember, experiment for learning—even an inconclusive result teaches us something!

How long to wait for conclusive results

- First, compare the potential impact of the experiment to the next experiment you'll run.
 - There are three key factors to consider:
 - **Visitors remaining:** this is an estimate of how long your experiment has to run to reach significance, based on traffic, conversion rate, and number of variations. Depending on how close you are, you might want to just wait it out; but if the number of visitors was exceeded, or if stat sig won't be reached any time soon, you may want to declare the experiment inconclusive and move on.
 - **Confidence interval:** this is useful when looking at the number of visitors remaining: for example, if the interval is $+.02\%-.04\%$ and there are still 6500 visitors remaining, is it worth it to keep going?
 - **Segment results:** you might find that stat sig was reached with one group, or maybe there's a single group that responded differently than others.