Customer obsession.

Invent and simplify

Why do you want to work at Facebook?

There are three main reasons:

1. I like FB's mission to help people build communities; I find that very people-centric which is aligned to how I am.
2. There is a lot of connectivity of different types at FB and I am sure that my knowledge of graphs analytics will have many applications.
3. I feel good when my work has a positive impact on millions or billions of people and I am always looking for these opportunities to think big and make a change.
4. Want to work with more data (in quantity and type).
5. Diverse in thought and technology.
6. Responsibility and taking care of customers is in DNA.
7. Interest in governance (offered pro bono help with bus scheduling in Plovdiv).

Failure (add noise)

- When was the last time you failed at work?
- Describe a project that you wish you had done better and how you would do it differently today

Situation: We got an escalation from an account manager, as the forecasting performance they promised to a client was not met. Moreover, some seasonal effects the client though should be there were not present.
Task: I was to investigate the event.
Action: I analyzed the impact of the issue and presented it to the team. The main reason was simply noise in the data. However, I got strong request from the account manager to add effects to the forecast that didn't improve the performance. This request was pushed by the CEO as

well.
Result: We didn't lose the customer, but the fee was reduced and we were 'on the bubble' for a bit.
Result: We did it for a critical meeting, requested by the account manager, despite my pleas to instead add confidence intervals etc. After that I had a long meeting with the account manager, CEO, and project management team. We prioritized creating forecast augmentation tool, allowing users (both internal and external), allowing them to modify the forecast, check confidence intervals and other metrics, and offering limited model exploration/explanations. This helped anyone to quickly assess the forecastability of a dataset, helping in project initiation phase. I did fail in communicating clearly and being proactive and making sure sales are prepared for the client meetings, are able to quickly come up with KPI's to agree on that are realistic, and have the proper tools to manage the expectation. Also, we did fail in providing enough tools for the customers to play around with incorporating their business knowledge into our solutions, as sometimes their data alone doesn't support predicting the quantity of interest.
Learning: I advised strongly a team member to always be present while presenting results. Absent that, we were at fault for not providing good enough tooling to customer-facing people, so that they can see the performance of what they're selling and we prioritized that over the next quarter.

Lead a team (FCMCF enchancements)

Situation: We were getting interpretability-related customer escalations on our optimization models very often. E.g. suggestions differed too much from their current MO. I was the developer of the optimization and it was up to spec. Investigations were very labor-intensive.
Task: I had to investigate for problems in the optimization, reduce the investigations, make for better customer experience
Action: I had a 1:1 with a representative of every customer and business analyst, and learned that the main bottleneck is the quality of input data, coming from the customer. I found about 6-7 systematic mistakes they make that confuse the model. I wrote several tools that show problems with data, clean it, visualize the solution and parts of it, and search within the space of possible related solutions. For example, one could change parts of the solutions and see where it's more costly than the original.
Result: Requests reduced in quantity, urgency, and time-to resolve. Particularly the time-to-respond and investigation speed was reduced about 10-fold. I first took upon myself the burden of about 1/3 of the customer requests and made sure the tooling I was writing was helping me do things faster, then share with the rest of the team, and finally with the customers.It's a difficult problem and the impact was high- we got 3 more customers and we don't have dedicated full or half- time people, serving customer inquiries any more.
Learning: I learned about the multiplying effect of proactively listening, creating better UI experience and dogfooding.

***

Conflict within a team (bwin combos)

Tell me about a time when you faced conflict within a team, and how you dealt with it (focus on team dynamics a bit more)

Situation: The company was releasing a new feature (combo bets) and we had to calculate it's price, including risk profile. Another scientist developed a simple and quick algorithm, which was easy to productize, but was not IMO faithfully representing the problem and thus representing potential loss of revenue. The team had almost decided to ship this solution.
Task: I had my own idea about the pricing model I wanted to check.
Action: I created a document describing a mathematical formulation of the problem, wrote a program, solving it. It was too slow for mass use in production. Figured out how the competing solution approximates the exact one. Wrote a program, examining the differences between the 2 across the expected input space. Figured out how to use the competing solution where it's very close to the exact one, and approximate better in the rest of the input space.
Result: We were confident we're delivering something that would maximize revenue. My framing allowed us to separate modeling (ideal state) and computation and show where exactly the two models meet. Showing the math formulation and that the competing solution would output nonsense sometimes, but is otherwise good approximation, made for everyone feel appreciated.
Learning: I learned about the importance of starting from an ideal mathematical model formulation. If one writes down the ideal situation and then makes simplifications, it's more explicit what trade-offs we make between business impact and other constraints (latency in this case).

Disagreed with Manager (pivo)

Describe a time you disagreed with your manager

Situation: Company was struggling with keeping disciplined between working towards product-market for existing products by developing both the product and the market, and customers with novel use-cases. Everyone (CEO, sales, eng, data science, marketing) has a

plausible multi-faceted story to tell. In particular, I was disagreeing about doing a particular new product, as the market seemed small and the pilot client was not reliable.
Task: I wanted to put discussions about the above in firmer ground.
Action: I wrote a tool, based on differential equations and optimal control, which would take some market research and dev planning data and decide if we should enter a market/ create a new product, and how: when to focus on customer vs product work, when to push to acquire new customers, etc.
I created a UI, I played around with some data for past projects/products and found it's reasonable. Distributed to Management board.
Result: I convinced the CEO and other members of the team to use the tool. They've been using it to decide on new projects, and when to push for acquiring new customers. Though parameters are very difficult to put in, it's been very useful in the past 3 years for estimating when a new product will go in the black, etc.
Learning: Making a trustworthy tool for everyone to experiment with shifted the focus of the conversation discussing characteristics of the market and product, rather than iterating on analysis.

Tell me about a time you overcame a really difficult challenge

Tell me about a time you lead a team

Mostly this, but focus more on the difficulty of the investigations and technical challenges.

Tell me a piece of difficult feedback you received and how you handled it

Situation: There was an several escalations a client, regarding forecasting accuracy. Thethe responsible data scientist in my team checked their work and it was fine. However, the account managers were not happy, escalated and I was called out for the team underperforming.
Task: Improve the customer outcome, as well as the relationship of team and account management.
Action: I went through the results with the data scientist, gathered notes, met account manager and customer, and heard their pain points. We looked at the particulars of the use case and created custom decision-making tools for the customer that worked with high level of noise.
Learning: We prioritized creating forecast augmentation tool, allowing users (both internal and external), allowing them to modify the forecast, check confidence intervals and other metrics, and offering limited model exploration/explanations. This helped anyone to quickly assess the

forecastability of a dataset, helping in project initiation phase, iterate faster, reduce uncertainty in what performance can be expected. It also helps both clients, the account managers, and us during meetings.

How do you get people to agree with your point of view?

I use two strategies:

1. Help people to detach themselves from the problem and their day-to-day struggles. Facilitate the conversation by sharing the ideal final state that we all wish to reach. From this shared understanding and alignment, it is easier to have more constructive conversations.Provide data to support arguments, while making sure many reasonable points of view are taken into account; try to quantify the impact of the effects each person takes to heart; get feedback if the data seems representative or someone's viewpoint is shorthanded.

Teamwork

Situation: The client had an operational transportation network, wanted to make some strategic changes. This was a consulting project, requirements were out of scope of our current products.
Task: Create a simulation framework on the basis of our current solution that would make realistic simulation under a variety of network scenarios.
Action: I created our first model and we showed the results to the customer. It was quite unrealistic in that the proposed changes in the network were drastic, compared to previous years. I partnered with our project and account manager during customer meetings and outside to gather data and narrow down what the current operations look like and the acceptable amount of operational changes to the customer operations. I worked with them to create a UI where it's easy to do such simulations.
Result: We had 5 scenarios we presented in the end, 'Greenfield', 'Baseline'-simulating current state of the ops, and 3 business scenarios w/ changes to network and working times. Clients were ecstatic and went on to implement one of the scenarios, involving opening 1 new branch, with an estimated saving of 20%
Learning: Often there's a large risk that requirements will be misspecified to some extent and it's only discovered when someone with intuition looks at it. If I perceive such a risk, I try to create some prototype tools as quickly as possible to facilitate the above process. Working with the project and account manager to that end made the project possible.

Invent and Simplify

See Describe a time you disagreed with your manager

I invented a new model, and simplified the discussions, moving from difference in both perception about market/dev time and logical reasoning, to just difference in perceptions about market characteristics.

Tell me about your most significant accomplishment. Why was it significant?

Tell me about a time you lead a team

It's a difficult problem and the impact was high- we got 3 more customers and we don't have dedicated full or half- time people, serving customer inquiries any more

---

Prioritization

When you're working with a large number of customers, it's tricky to deliver excellent service to them all. How do you go about prioritizing your customers' needs?

I'll work on the highest impact-urgency item if there's an estimate for the impact.

In terms of priority, I try to balance between impact, deadlines and probability to succeed within a deadline.

Unblocking other people: For example, often a baseline analytical product would be relatively quick to make, and also an MVP grounds the project on solid ground. E.g. I will try to make a prototype, even with small and/or fake data (very applicable for OR type problems). When this is done, I might switch if the project is blocked in some other way (e.g. data acquisition).

I also like to spend some time thinking about generalizing done to other domains (e.g. delivery planning and container shipping), synergies and so on.

On a daily basis, I tend to give priority to tasks that unblock others and tasks that only I can do much faster/better than others, or ones that keep . Otherwise, I use a mix of urgency and importance.
One exception is if my current task is very complicated and difficult to checkpoint. In that case I'd rather make progress with it. But this is risky, so I try to get better at ‘checkpointing’ well and leaving things at a status where it’s clear what the next step is.

Didn’t Finish

Tell me about a time when you had to leave a task unfinished.

Situation: Large account/Very high impact forecasting project, weren’t close to the desired accuracy, big problem with some of the holidays. Forecast for whole pacific ocean. Had only data for 2019-2021 of data, so very tricky to figure out yearly seasonality.
Task: Entered the project 10 days before end urgently, from paternity leave, try to improve holiday performance in particular.
Action: Created some tools so it’s easy to query data around different holidays. Manually looked trough problematic holidays, no apparent patterns. Pulled some old external trading data for 2008-. Did some normalization tricks to put the residuals of the 2 datasets together, and then in some places the relationship was apparent. Quickly did some simple linear regressions to adjust holiday weeks, based on the whole period, and added them to the teammate’s models as post-processing.
Result: We had 4% improvement on the backtesting error, and managed to win the account, but more importantly about half of the error outliers were much smoother. The tricks were very crude and both introducing some bias, and missing a lot of detail about the holiday effects.With more work we’ll be able to provide a better forecast, and we’ll soon have the opport.
Learning: Sometimes it’s good to take even imperfect data, and eyeballs for patterns there.I could learn faster on the challenges of that attribute and pivot quickly to another attribute for which we have more chances to succeed.

Comment on didn’t finish:

The most important things are to

1. do the 20% of most impactful work and have something useable
2. Checkpoint+document so it's clear what next steps would be.

'Old stories' (no STARL format/unstructured), might drop them

• Tell the story of the last time you had to apologize to someone.(don't tell)

We had a deadline for developing a certain optimization module for a new customer a week ago. I did the module, reviewed the results briefly and send it over to the product owner. She also did review it and we handed the docker over to the implementation team. However, had I ran and reviewed the simulation for a longer time, I would have noticed a bug. The bug appeared only in certain high-demand situations (network close to capacity, I was creating some duplicated trucks, but then had the same ID and were thus re-used).

The product owner also didn't notice the irregularity until we ran the thing on the engineering server. So we had to deploy the hotfix and missed the billing period. I apologized to the product owner and the president, explained the cause of the bug and why it wasn't caught on my end and discussed together more thorough test case coverage.

Test cases for this product are tricky to engineer.

• Who was your most difficult customer?

- Most difficult customer
  KN imputation task- problem is there is both 'no data' and they didn't believe in what we are doing. Thus various requests about data definition would take quite a bit longer than the usual to finish. In the end:
  -- Did a custom model, combination of several rule-based and statistical imputation techniques, chained together.
  --- Gathered data by spot-checks (on customer's site, in another country, etc) and did spot checks with the models, balancing the normal error measures on piece level vs the global thing.
  --- Did presentations to multiple teams - operational team, focused on 'correcntess' of predictions of something they thought immesurable from their data.
  ---- Presentation to technical team
  --- Customer sponsor of project changed and we had to change the whole problem in order to continue the project.

• Give me an example of a time when you did not meet a client's expectation. What happened, and how did you attempt to rectify the situation?

Often we 'sell' forecasts with a promise about a particular accuracy measure. We check statistically, perhaps with several models, the estimated levels of noise, etc.

We double check we did the job 'by the book' so we're not missing some silly errors. Then think of how to get other signals (e.g. upcoming change of behavior in some of the client's client) and make it possible to integrate them into the forecast. Also check different models, etc.

Focus on the ‘human-in-the-loop’ part of the solution: knobs and levers, priors if there are clear signals the experts use we might not catch, etc.

** Try to convince first ourselves and then the customer that this is roughly the noise level, implied by the data,

** think creatively what additional theory (expert knowledge) and data can be added to reduce it

** (some levers and knobs for the expert user to be able to play it so they still can take better decisions).

Problem of measurability.

Tell me about a time when you had to work on a project with unclear responsibilities/ Bias for action

There was a large project about guessing how full the customer’s trucks are. That was it. We had very incomplete data about package sizes and we just had to make it work so the customer believes in it

There was a time when I was asked to consult on a new customer project, helping one of our lead data scientists with it. There was a concept of the product, but nothing substantial. I took up on the task to facilitate both the initial meeting and the implementation as much as possible.

I wasn’t going to attend the meeting myself, as I was ill.

I did:

Discuss with the sales people the customer pain point, the solution they currently have and their proposed current solution.
Write down a linear program, taking the data I imagined and optimizing the metric in question.

Called a meeting beforehand, explaining the concept, what the output might look like and what input data is required. The idea was to ground/frame the discussion.
There were two days of meetings. After the first one, I had a call with my teammate, discussing what part of the plan matched their expectations and what needed to be modified. We re-visited the deck for the next day. That day we didn't have major changes, so at the point I

• Tell me about a time when you invented something.

Waremetrics story.

Hire and develop the best.

We had maybe about 5-6 people. I adhere to the following principles:
1. Whoever does the bulk of the work will present it if he/she requests.
If there are worries about their inexperience with customer calls, we do mock calls and presentations and simply pay more attention to both the analysis and the story.
