

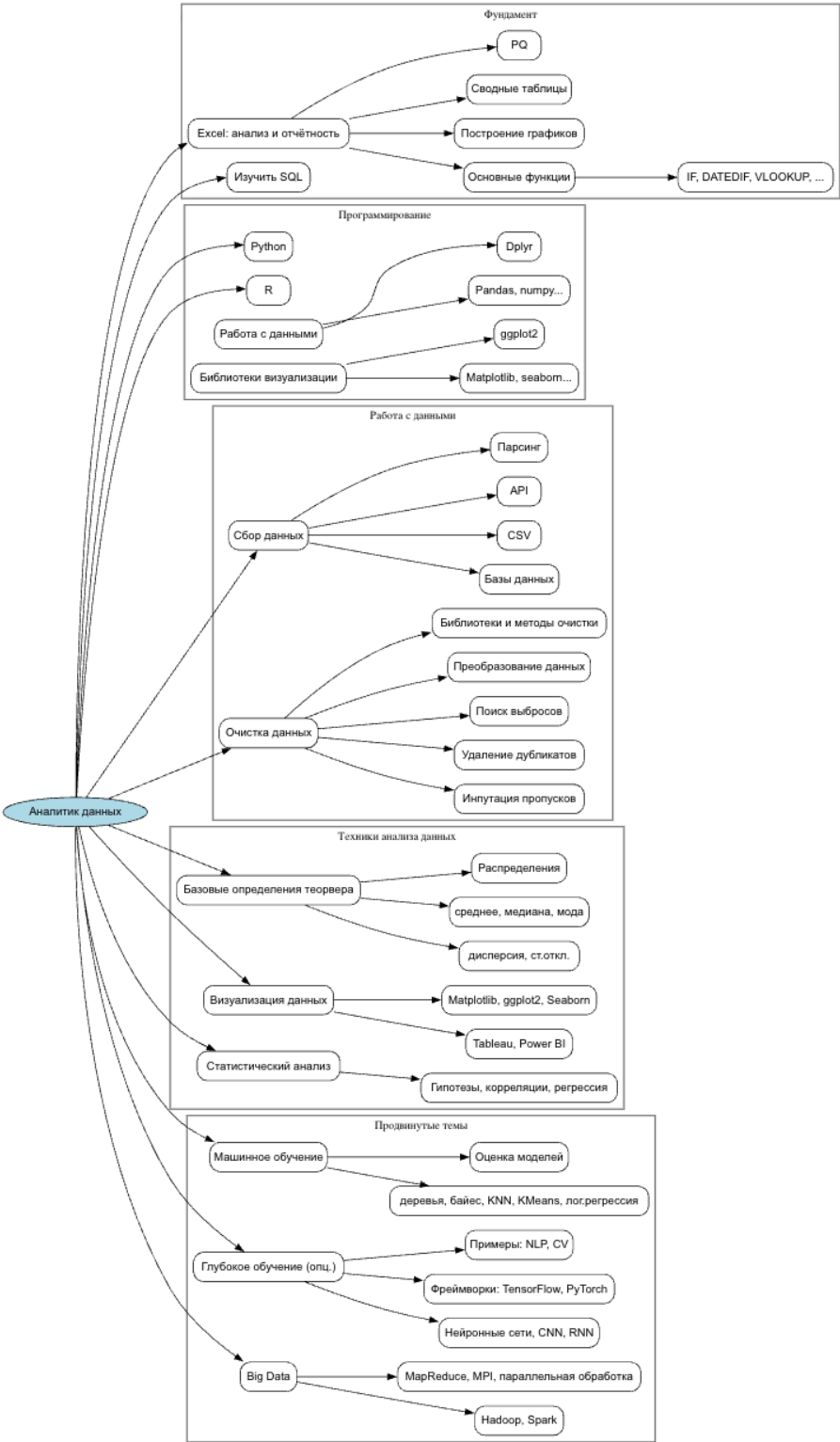
# Рoadмaп aналитикa дaнных: пoдpoбнoe pyкoвoдствo пo oбyчeнию

21 сeнтябpя 2025 г.

## Сoдepжaниe

<b>1</b>	<b>Фундамент</b>	<b>4</b>
1.1	Excel: анализ и отчётность (включая Power Query)	4
1.2	Изучить SQL	4
<b>2</b>	<b>Программирование</b>	<b>4</b>
2.1	Python	4
2.2	R	4
<b>3</b>	<b>Работа с данными</b>	<b>4</b>
3.1	Сбор данных	4
3.2	Очистка данных	5
<b>4</b>	<b>Техники анализа данных</b>	<b>5</b>
4.1	Визуализация и сторителлинг	5
4.2	Статистический анализ: базовые определения	5
<b>5</b>	<b>Статистика и эконометрика: приоритеты и углубление</b>	<b>5</b>
5.1	Приоритетные методы (обязательный минимум)	5
5.2	Продвинутые методы для аналитика	5
5.3	A/B-тестирование и эксперименты	6
5.4	Гайды и книги по статистике/эконометрике	6
<b>6</b>	<b>Продвинутые темы</b>	<b>6</b>
6.1	Машинное обучение (обзор для аналитика)	6
6.2	Глубокое обучение (опционально)	6
6.3	Big Data и аналитическая инженерия	6
<b>7</b>	<b>Пет-проекты</b>	<b>6</b>
7.1	1. Прогнозирование временных рядов	6
7.2	2. Проектирование базы данных SQL	7
7.3	3. Когортный анализ удержания клиентов	7
7.4	4. Корреляционный анализ	7
7.5	5. Работа с данными	8
7.6	6. Дашборды	8
7.7	7. A/B-тест	8

7.8	8. Пайплайн	8
7.9	9. Система алертов	9



## 1 Фундамент

### 1.1 Excel: анализ и отчётность (включая Power Query)

**Что учить:** функции IF, XLOOKUP/VLOOKUP, INDEX+MATCH, SUMIF/COUNTIF, TEXTJOIN/CONCAT, LEFT/RIGHT/MID, TRIM, SUBSTITUTE, UPPER/LOWER/PROPER, DATEDIF, работа с датами/временем. Сводные таблицы: группировка дат, вычисляемые поля, срезы и таймлайны, связь с моделью данных. Графики: столбчатые, линейные, комбинированные, использование второй оси, подписи, форматирование шкал. Power Query (PQ): импорт CSV/Excel/SQL, типы данных, Merge/Append, параметризация, обновление и refresh chain.

**Практика:** просто ищите сырые датасеты (CSV → PQ → сводная → графики → PDF).

**Ресурсы:** [Microsoft Excel Support](#), [Power Query Docs](#), [DAX Studio](#) (для модели данных).

### 1.2 Изучить SQL

**Что учить:** база: SELECT, WHERE, ORDER BY, LIMIT, CASE WHEN. Соединения: INNER/LEFT/RIGHT/FULL JOIN, USING/ON. Агрегации: GROUP BY, HAVING, COUNT/SUM/AVG/MIN/MAX. Подзапросы и CTE (WITH); оконные функции: ROW\_NUMBER, RANK, LAG/LEAD, SUM() OVER. Даты, строки, регулярные выражения (где доступны); планы выполнения (EXPLAIN) и индексы — базово.

**Практика:** 50–100 задач с [sq-ex.ru](#) (магазин/подписки).

**Ресурсы:** [Mode SQL Tutorial](#), [PostgreSQL Docs](#), [SQL Style Guide](#).

## 2 Программирование

### 2.1 Python

**Что учить:** pandas (индексация, merge/join, groupby, pivot, даты), NumPy, визуализация (matplotlib, seaborn), работа с БД (SQLAlchemy), API (requests), проекты в Jupyter.

**Ресурсы:** [pandas docs](#), [NumPy docs](#), [Matplotlib docs](#), [Seaborn docs](#), [requests docs](#), [SQLAlchemy](#), [python course](#).

### 2.2 R

**Что учить:** tidyverse — dplyr/tidyr (pipes, mutate, summarise, join), ggplot2, отчёты в RMarkdown/Quarto.

**Ресурсы:** [R for Data Science \(2e\)](#), [dplyr](#), [ggplot2](#).

## 3 Работа с данными

### 3.1 Сбор данных

**Источники:** базы данных (PostgreSQL/MySQL/SQL Server), файловые форматы (CSV/Parquet), API, парсинг (с учётом robots.txt и правил сайта).

**Навыки:** аутентификация, пагинация, лимиты запросов, бэкофф/ретрай, логирование загрузок, сохранение «сырых» слоёв.

Ресурсы: [REST basics](#), [MDN HTTP](#), [requests](#), [pandas I/O](#).

### 3.2 Очистка данных

**Что делать:** импутация пропусков (среднее, медиана, мода, forward/backfill, флаг пропусков как признак). Дубликаты: ключи и составные ключи, дедупликация оконными функциями. Выбросы: 1.5 IQR,  $z$ -score, контекстные правила бизнеса. Преобразования: типы данных, нормализация категорий, разбиение и склейка столбцов, wide↔long. Валидация: тесты диапазонов и типов, базовые data quality чек-листы.

Ресурсы: [pandas](#), [dplyr](#), [Great Expectations \(data quality\)](#).

## 4 Техники анализа данных

### 4.1 Визуализация и сторителлинг

**Что изучать:** инструменты Tableau и Power BI для дашбордов; matplotlib, ggplot2 и seaborn для аналитических графиков. Типы диаграмм: столбчатые, линейные, гистограммы, плотности, боксплоты, scatter, heatmap, funnel, treemap. Круговые — ограниченно. Принципы: одна диаграмма — один тезис; читаемая легенда; корректные шкалы и единицы; аннотации ключевых точек.

Ресурсы: [Fundamentals of Data Visualization](#), [Storytelling with Data](#), [Power BI Learn](#), [Tableau Training](#).

### 4.2 Статистический анализ: базовые определения

Случайные величины, выборка и генеральная совокупность, закон больших чисел, ЦПТ; описательные метрики (среднее, медиана, мода; дисперсия, стандартное отклонение; перцентили); виды распределений (нормальное, Бернулли, биномиальное, Пуассон).

## 5 Статистика и эконометрика: приоритеты и углубление

### 5.1 Приоритетные методы (обязательный минимум)

Проверка гипотез и интервалов:  $z$ - и  $t$ -тесты (в том числе Welch), доверительные интервалы, проверка предпосылок (нормальность, равенство дисперсий). Непараметрические тесты: Mann–Whitney U, Wilcoxon signed-rank,  $\chi^2$  на независимость и согласие. ANOVA/ANCOVA: сравнение нескольких групп, post-hoc (Tukey), ковариаты. Корреляции: Пирсон и Спирмен, отличие корреляции от причинности. Регрессия OLS: постановка, предпосылки (линейность, независимость, гомоскедастичность, нормальность остатков), диагностика (тест Бреуша–Пагана, автокорреляция Дурбина–Уотсона). Эффект-размер и мощность: Cohen's  $d$ , AUC; расчёт мощности и MDE для A/B. Множественные сравнения: контроль (Bonferroni, Holm, Benjamini–Hochberg).

### 5.2 Продвинутое методы для аналитика

GLM: логистическая регрессия (классификация), Пуассон и негативная биномиальная модели для счётных данных. Робастные оценки: HC0–HC3 робастные стандартные ошиб-

ки, бутстрэп, перестановочные тесты. Временные ряды: стационарность (ADF/KPSS), сезонность, автокорреляции (ACF/PACF), модели ARIMA/SARIMA/ETS, кросс-валидация по времени, прогнозирование точек и интервалов. Панельные данные: фиксированные и случайные эффекты, тест Хаусмана, кластерные ошибки. Каузальный анализ. Иерархические и смешанные модели: случайные перехваты и наклоны. Байесовские методы: априоры, апостериоры, MAP/HPD интервалы, MCMC.

### 5.3 А/В-тестирование и эксперименты

Дизайн: рандомизация, стратификация, блокировка, размер выборки и мощность, MDE. Анализ: ITT и PP, непараметрические альтернативы, доверительные интервалы эффекта, sequential testing (Alpha spending). Метрики: выбор основной метрики, сезонность и календарные эффекты.

### 5.4 Гайды и книги по статистике/эконометрике

OpenIntro Statistics<sup>1</sup>, ISLR<sup>2</sup>, FPP3<sup>3</sup>, Causal Inference: The Mixtape<sup>4</sup>, Angrist & Pischke *Mastering 'Metrics*, Wooldridge *Introductory Econometrics*.

## 6 Продвинутые темы

### 6.1 Машинное обучение (обзор для аналитика)

Алгоритмы: деревья и ансамбли, наивный Байес,  $k$ NN,  $k$ -means, логистическая регрессия. Оценка: holdout и кросс-валидация, калибровка, важность признаков, leakage, стабильность во времени. Ресурсы: scikit-learn, XGBoost, CatBoost.

### 6.2 Глубокое обучение (опционально)

DNN, CNN, RNN/Seq2Seq, эмбединги; фреймворки TensorFlow/Keras, PyTorch. Ресурсы: TensorFlow, PyTorch.

### 6.3 Big Data и аналитическая инженерия

Spark (DataFrame API, Spark SQL), облачные DWH (BigQuery/Snowflake/Redshift), оркестрация (Airflow), моделирование dbt.

## 7 Пет-проекты

### 7.1 1. Прогнозирование временных рядов

На более старших позициях аналитикам часто нужно лезть в ML/Эконометрику, особенно риск-аналитикам, поэтому нужно будет заботиться обо этих навыках. В этом проекте будем прогнозировать спрос на следующий месяц и выявлять факторы воздействия на точность прогноза. Возьмём датасет продаж/курс валют/трафик веб-сайта и т.п.

<sup>1</sup><https://www.openintro.org/book/os/>

<sup>2</sup><https://www.statlearning.com/>

<sup>3</sup><https://otexts.com/fpp3/>

<sup>4</sup><https://mixtape.scunning.com/>

Работать будем в `Jupyter` или коллабе; если датасет большой, то стоит подключить `CUDA`. Очистим данные от выбросов, проанализируем ряд на тренд, остаток и сезонность. Многие модели требуют стационарности, поэтому её нужно протестировать; если ряд не стационарный, то применяем дифференцирование и/или логарифмирование. В качестве базовых моделей используем `ARIMA` и `Prophet`. Далее оцениваем качество прогноза (например, по `WAPE`) и формулируем выводы.

## 7.2 2. Проектирование базы данных SQL

`SQL` — довольно простой язык, и с помощью одного пет-проекта можно существенно продвинуться за неделю. Для начала выбираем тематику БД (например, интернет-магазин). Первый этап — проектирование: составляем концептуальную модель (только названия сущностей и связи), затем логическую и физическую модели. В физической модели описываем поля сущностей, ограничения и типы; в логической — рисуем схему связей (по каким ключам связь, тип ключа и тип связи), можно описать и стратегию версионирования. Реализацию можно вести в `VS Code` (потребуется первичная настройка) либо в `DBeaver`: поднимаем `PostgreSQL/pgAdmin`, создаём БД, подключаемся из `DBeaver`. Далее пишем `DDL`-скрипты строго по логической и физической моделям (по пути изучаем индексы и расставляем их). После этого реализуем `DML` — генерацию данных и запросы. Стараемся задействовать продвинутый функционал: заполнение случайными данными, процедуры/представления, запросы с оконными функциями, `CTE` и, при желании, рекурсией.

## 7.3 3. Когортный анализ удержания клиентов

Цель проекта — показать именно продуктивное понимание: собрать как можно больше выводов и гипотез, а не демонстрировать «харды». Выгружаем транзакции какого-нибудь ритейлера. Когорты формируем по месяцу первой покупки. Рассчитываем долю клиентов, совершающих повторные покупки на 7-й, 30-й и 90-й дни. Строим тепловую карту удержания по когортам. На такой карте хорошо видны сезонные эффекты: например, декабрьские когорты часто показывают более низкое удержание из-за новогоднего ажиотажа и разовых покупок подарков. Также можно заметить выбросы, обусловленные маркетинговыми акциями и другими событиями. Далее сегментируем когорты по каналу привлечения (контекстная реклама/соцсети) и ищем различия между сегментами. В выводах анализируем общий тренд удержания, факторы, влияющие на него, и эффективность каналов.

## 7.4 4. Корреляционный анализ

Сначала выбираем тему и данные: подбираем две или больше числовых характеристик, где предполагается связь — например, «влияет ли размер скидки на объём продаж» или «связь времени учебы и оценки». Можно взять готовый датасет с `Kaggle` либо сгенерировать/спарсить собственные данные. Проводим исследовательский анализ (`EDA`) и визуализируем связи. Считаем коэффициенты: Пирсона — для линейных связей и приблизительно нормальных данных; Спирмена — для любых монотонных связей через ранги; Кендалла — как альтернативу Спирмену на небольших выборках. Для Пирсона предварительно проверяем нормальность (Шапиро–Уилка или `Q-Q plot`). Оцениваем статистическую значимость через  $p$ -value. В выводах описываем тип связи и её значимость; важно не путать корреляцию с причинностью. Дополнительно можно провести регрессионный анализ и отобразить линию регрессии на графике.

## 7.5 5. Работа с данными

В новом продукте придётся выстраивать аналитические процессы, но сперва нужно привести данные к удобному виду. Хранилище может быть не оптимальным: например, есть база с одной таблицей, где хранятся и посты, и авторы. Поскольку постов гораздо больше, разумно вынести их в отдельную таблицу, а исходную нормализовать. Кроме неэффективного хранения, данные бывают «грязными»: аномалии, дубликаты, пропуски, неудобные типы. Для очистки пригодятся статистические методы и визуализация. Уже на этом этапе можно формулировать гипотезы. Для выполнения такого задания достаточно открыть **Jupyter Notebook**, взять любой «сырой» датасет (например, с Kaggle), имитирующий «большие данные», и ориентироваться на примеры EDA от других пользователей.

## 7.6 6. Дашборды

Результаты предыдущей работы необходимо представить в наглядном и доступном виде. К ключевым показателям продукта должен быть быстрый доступ всей команды; странно каждый раз писать запрос, чтобы узнать, сколько у продукта пользователей. Поэтому аналитика часто начинается с вывода базовых метрик (DAU, WAU, MAU). Целевая задача — визуализация и презентация; уместно придумывать собственные метрики и формулировать гипотезы, глядя на графики. Например, заметили пик активных пользователей (маркетинг закупил рекламу), а затем отток — изучаем retention и оцениваем эффективность кампании. Для окружения можно использовать ClickHouse, Redash, Superset и GitLab: инструменты интерактивные, с хорошими туториалами. На работе набор инструментов может отличаться, но их освоение не составит труда.

## 7.7 7. А/В-тест

Здесь пригодятся гипотезы из предыдущих проектов — теперь их можно проверить экспериментально. Сначала планируем эксперимент: оцениваем необходимое количество пользователей, выбираем метод проверки. Пространства для творчества много: можно подбирать более чувствительные метрики, тестировать систему сплитования, использовать методы понижения дисперсии. Для начала достаточно выбрать пользователей, разделить их на тест и контроль и применить  $t$ -тест или Манна–Уитни, затем корректно интерпретировать результат и интегрировать всё в рабочее окружение из проекта про дашборды. По мере готовности добавляем хеширование с солью, АА-тест, бутстрэп, CUPED, бакетное преобразование и т. д. Примеры А/В-тестов разного качества легко найти на Kaggle и GitHub.

## 7.8 8. Пайплайн

Данные обычно хранятся в разных системах и форматах, поэтому аналитику часто приходится переносить их в единое хранилище и выдавать результаты в виде графиков и таблиц. Для имитации такого процесса можно взять данные из одной базы (или потока, например Kafka), положить их в Hadoop, затем загрузить в другую базу, применив преобразования в Spark, и запустить весь процесс через Airflow. После такой практики элементы дата-инженерии станут гораздо понятнее.

## 7.9 9. Система алертов

В работе часто требуется регулярная отчётность, поэтому её удобно автоматизировать через Telegram-бота. Создаём бота, пишем скрипт, который собирает отчёт по выбранной БД. Продумываем метрики, период и формат представления; автоматизируем отправку отчёта с помощью Airflow. В дополнение можно реализовать детектирование аномалий: выбирать метрики, срезы и частоту мониторинга, определять метод детекта. Подходы делятся на статистические (правило трёх сигм) и ML-методы (DBSCAN, LOF). Начинать стоит с самого простого решения.