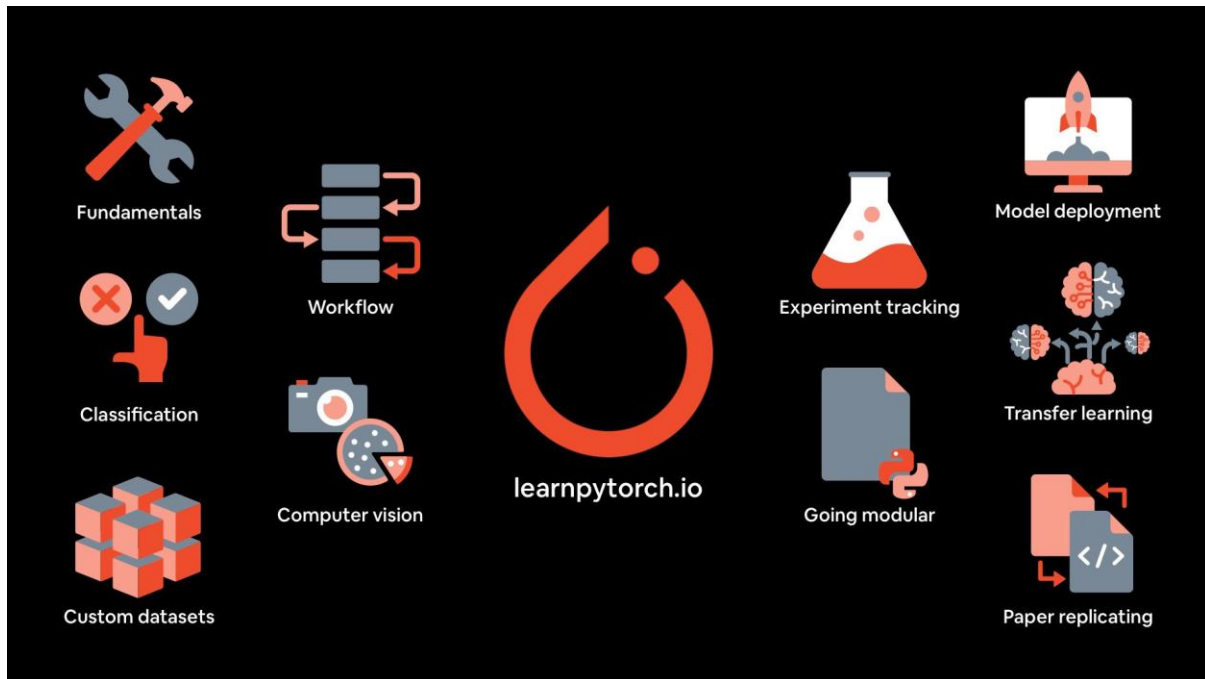


## Pelajari PyTorch untuk *Deep Learning*: Buku Zero to Mastery



00 - Dasar-dasar PyTorch	Banyak operasi dasar PyTorch yang digunakan untuk <i>Deep Learning</i> dan jaringan saraf.
01 - Alur Kerja PyTorch	Memberikan garis besar untuk mendekati masalah <i>Deep Learning</i> dan membangun jaringan saraf dengan PyTorch.
02 - Klasifikasi Jaringan Syaraf PyTorch	Menggunakan alur kerja PyTorch dari 01 untuk mengatasi masalah klasifikasi jaringan saraf.
03 - Visi Komputer PyTorch	Mari kita lihat bagaimana PyTorch dapat digunakan untuk masalah computer vision menggunakan alur kerja yang sama dari 01 & 02.

### 00. PyTorch Fundamentals

PyTorch adalah kerangka kerja pembelajaran mesin sumber terbuka yang digunakan untuk memanipulasi data dan menulis algoritma pembelajaran mesin.

Selanjutnya, kita akan mempelajari tensor, komponen dasar *Deep Learning*.

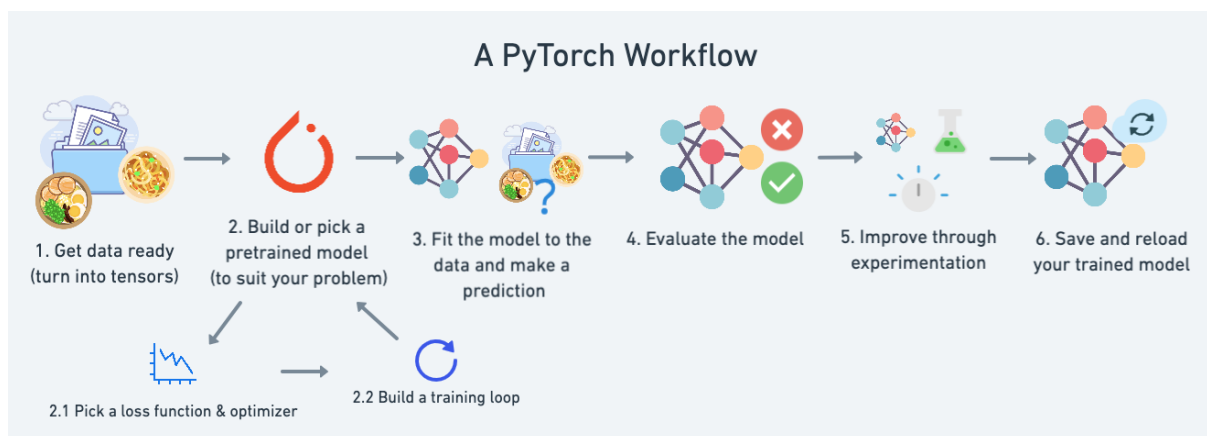
Tensor digunakan untuk menggambarkan data numerik, yang dapat berupa gambar, teks, atau tabel angka.

Menemukan pola dalam data juga merupakan tujuan umum pembelajaran mesin.

Oleh karena itu, pengetahuan tentang pembuatan, interaksi, dan manipulasi tensor sangat penting.

Tema	Isi
Pengantar tensor	Tensor adalah elemen dasar dari semua pembelajaran mesin dan pembelajaran mendalam.
Membuat tensor	Tensor dapat mewakili hampir semua jenis data (gambar, kata, tabel angka).
Mendapatkan informasi dari tensor	Jika Anda dapat memasukkan informasi ke dalam tensor, Anda pasti ingin mengeluarkannya juga.
Memanipulasi tensor	Algoritma pembelajaran mesin (seperti jaringan saraf) melibatkan manipulasi tensor dengan berbagai cara seperti menambah, mengalikan, menggabungkan.
Berurusan dengan bentuk tensor	Salah satu masalah paling umum dalam pembelajaran mesin adalah menangani ketidakcocokan bentuk (mencoba mencampurkan tensor berbentuk salah dengan tensor lain).
Pengindeksan pada tensor	Jika Anda telah mengindeks daftar Python atau array NumPy, ini sangat mirip dengan tensor, hanya saja mereka dapat memiliki dimensi yang jauh lebih besar.
Mencampur tensor PyTorch dan NumPy	PyTorch bermain dengan tensor (torch.Tensor), NumPy menyukai array (np.ndarray) terkadang Anda ingin memadupadankannya.
Reproduksibilitas	Pembelajaran mesin sangat eksperimental dan karena menggunakan banyak keacakan untuk bekerja, terkadang Anda ingin keacakan tersebut tidak terlalu acak.
Menjalankan tensor pada GPU	GPU (Graphics Processing Unit) membuat kode Anda lebih cepat, PyTorch memudahkan menjalankan kode Anda di GPU.

## 01. PyTorch Workflow Fundamentals



Tema	Isi
1. Mempersiapkan data	Data bisa berupa apa saja, tetapi untuk memulainya kita akan membuat garis lurus sederhana
2. Membangun model	Di sini kita akan membuat model untuk mempelajari pola dalam data, kita juga akan memilih fungsi kerugian , pengoptimal , dan membangun loop pelatihan .
3. Menyesuaikan model dengan data (pelatihan)	Kita punya data dan model, sekarang biarkan model (mencoba) menemukan pola dalam data ( pelatihan ).
4. Membuat prediksi dan mengevaluasi suatu model (inferensi)	Model kita menemukan pola dalam data, mari kita bandingkan temuannya dengan data aktual ( pengujian ).
5. Menyimpan dan memuat model	Anda mungkin ingin menggunakan model Anda di tempat lain, atau kembali lagi nanti, di sini kami akan membahasnya.
6. Menyatukan semuanya	Mari kita ambil semua hal di atas dan gabungkan.

## 02. PyTorch Neural Network Classification

**"Is this email spam or not spam?"**

To: [daniel@mrdbourke.com](mailto:daniel@mrdbourke.com)  
Hey Daniel,

This deep learning course is incredible!  
I can't wait to use what I've learned!

**Not spam**

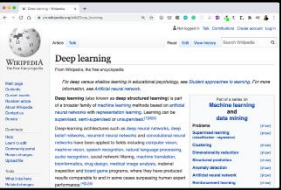
To: [daniel@mrdbourke.com](mailto:daniel@mrdbourke.com)  
Hay daniel...

C00ngratulations! U win \$1139239230

**Spam**


**Binary classification**  
(one thing or another)

**"What tags should this article have?"**



**Multilabel classification**

**"Is this a photo of sushi, steak or pizza?"**



**Multiclass classification**  
(more than one thing or another)

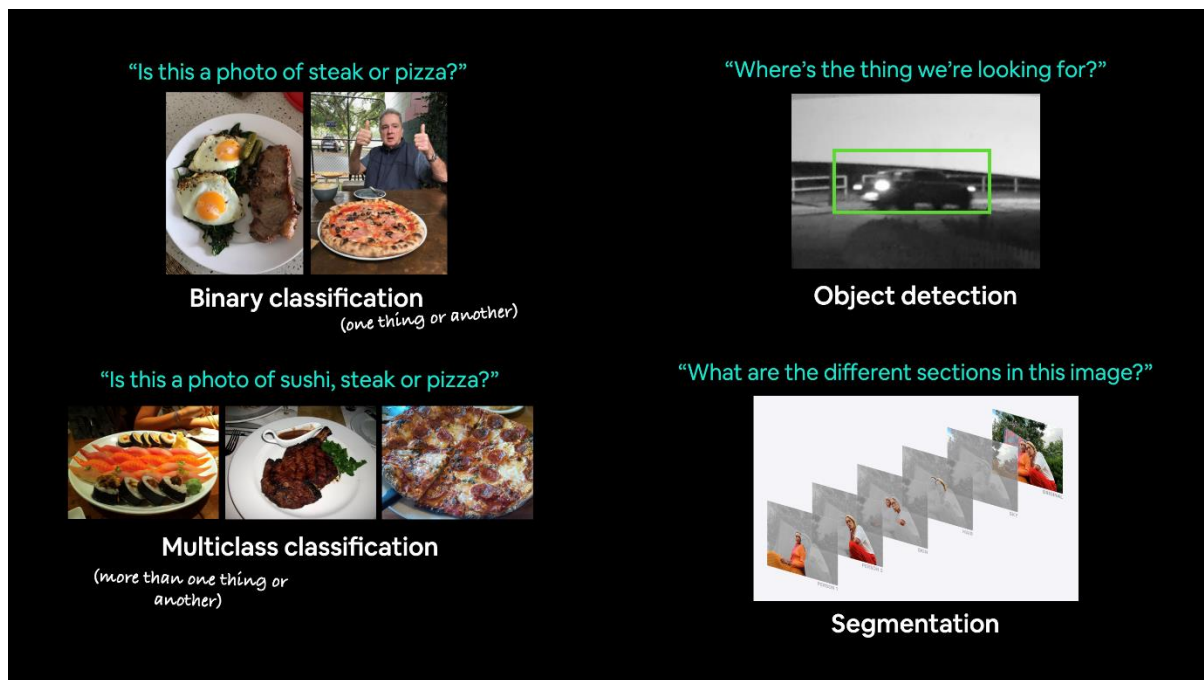
Klasifikasi memiliki prediksi.

Jenis masalah	Apa itu?
Klasifikasi biner	Target dapat berupa salah satu dari dua pilihan, misalnya ya atau tidak
Klasifikasi jamak	Target dapat berupa salah satu dari lebih dari dua pilihan
Klasifikasi multi-label	Target dapat diberikan lebih dari satu opsi

Tema	Isi
0. Arsitektur jaringan saraf klasifikasi	Jaringan saraf dapat hadir dalam hampir semua bentuk dan ukuran, namun biasanya mengikuti denah lantai yang serupa.
1. Mempersiapkan data klasifikasi biner	Data bisa berupa apa saja, tetapi untuk memulai kita akan membuat kumpulan data klasifikasi biner sederhana.
2. Membangun model klasifikasi PyTorch	Di sini kita akan membuat model untuk mempelajari pola dalam data, kita juga akan memilih fungsi kerugian , pengoptimal , dan membangun loop pelatihan khusus untuk klasifikasi.
3. Menyesuaikan model dengan data (pelatihan)	Kita punya data dan model, sekarang biarkan model (mencoba) menemukan pola dalam data ( pelatihan ).
4. Membuat prediksi dan mengevaluasi suatu model (inferensi)	Model kita menemukan pola dalam data, mari kita bandingkan temuannya dengan data aktual ( pengujian ).
5. Memperbaiki model (dari perspektif model)	Kami telah melatih model yang dievaluasi tetapi tidak berhasil, mari coba beberapa hal untuk memperbaikinya.
6. Non-linearitas	Selama ini model kita hanya mampu memodelkan garis lurus, bagaimana dengan garis non linier (tidak lurus)?
7. Mereplikasi fungsi non linier	Kami menggunakan fungsi non-linier untuk membantu memodelkan data non-linier, namun seperti apa bentuknya?
8. Menyatukan semuanya dengan klasifikasi kelas jamak	Mari kita gabungkan semua yang telah kita lakukan sejauh ini untuk klasifikasi biner dengan masalah klasifikasi kelas jamak.

### 03. PyTorch Computer Vision

Pembuatan model untuk mengklasifikasikan apakah suatu foto adalah kucing atau anjing ( klasifikasi biner). Atau apakah foto itu berupa kucing, anjing, atau ayam ( klasifikasi kelas jamak ). Atau mengidentifikasi di mana mobil muncul dalam bingkai video ( deteksi objek ). Atau mencari tahu di mana objek yang berbeda dalam suatu gambar dapat dipisahkan (segmentasi panoptik ).



Tema	Isi
0. Perpustakaan visi komputer di PyTorch	PyTorch memiliki banyak perpustakaan visi komputer bawaan yang bermanfaat, mari kita periksa.
1. Memuat data	Untuk melatih visi komputer, kita akan mulai dengan beberapa gambar pakaian berbeda dari FashionMNIST .
2. Siapkan datanya	Kita punya beberapa gambar, mari muat dengan PyTorchDataLoader sehingga kita bisa menggunakannya dengan loop pelatihan kita.
3. Model 0: Membangun model dasar	Di sini kita akan membuat model klasifikasi kelas jamak untuk mempelajari pola dalam data, kita juga akan memilih fungsi kerugian , pengoptimal, dan membangun loop pelatihan .
4. Membuat prediksi dan mengevaluasi model 0	Mari buat beberapa prediksi dengan model dasar kita dan evaluasi.
5. Siapkan kode agnostik perangkat untuk model masa depan	Praktik terbaiknya adalah menulis kode tanpa perangkat, jadi mari kita siapkan.
6. Model 1: Menambahkan non-linearitas	Bereksperimen adalah bagian besar dari pembelajaran mesin, mari kita coba dan tingkatkan model dasar kita dengan menambahkan lapisan non-linier.
7. Model 2: Jaringan Neural Konvolusional (CNN)	Saatnya untuk memperjelas visi komputer dan memperkenalkan arsitektur jaringan saraf konvolusional yang kuat.
8. Membandingkan model kami	Kami telah membuat tiga model berbeda, mari kita bandingkan.

9. Mengevaluasi model terbaik kita	Mari kita membuat beberapa prediksi pada gambar acak dan mengevaluasi model terbaik kita.
10. Membuat matriks konfusi	Matriks konfusi adalah cara terbaik untuk mengevaluasi model klasifikasi, mari kita lihat bagaimana kita dapat membuatnya.
11. Menyimpan dan memuat model dengan performa terbaik	Karena kita mungkin ingin menggunakan model kita untuk nanti, mari kita simpan dan pastikan model dimuat kembali dengan benar.